Nr. 6/86 Juni
DM 6.50, sfr 6.50, öS 50, Lit 5900, hfl 7.50

peeker

Funktionsplotter in Double-Hires-Grafik

Megaboard- und AFDC-Massenspeicher

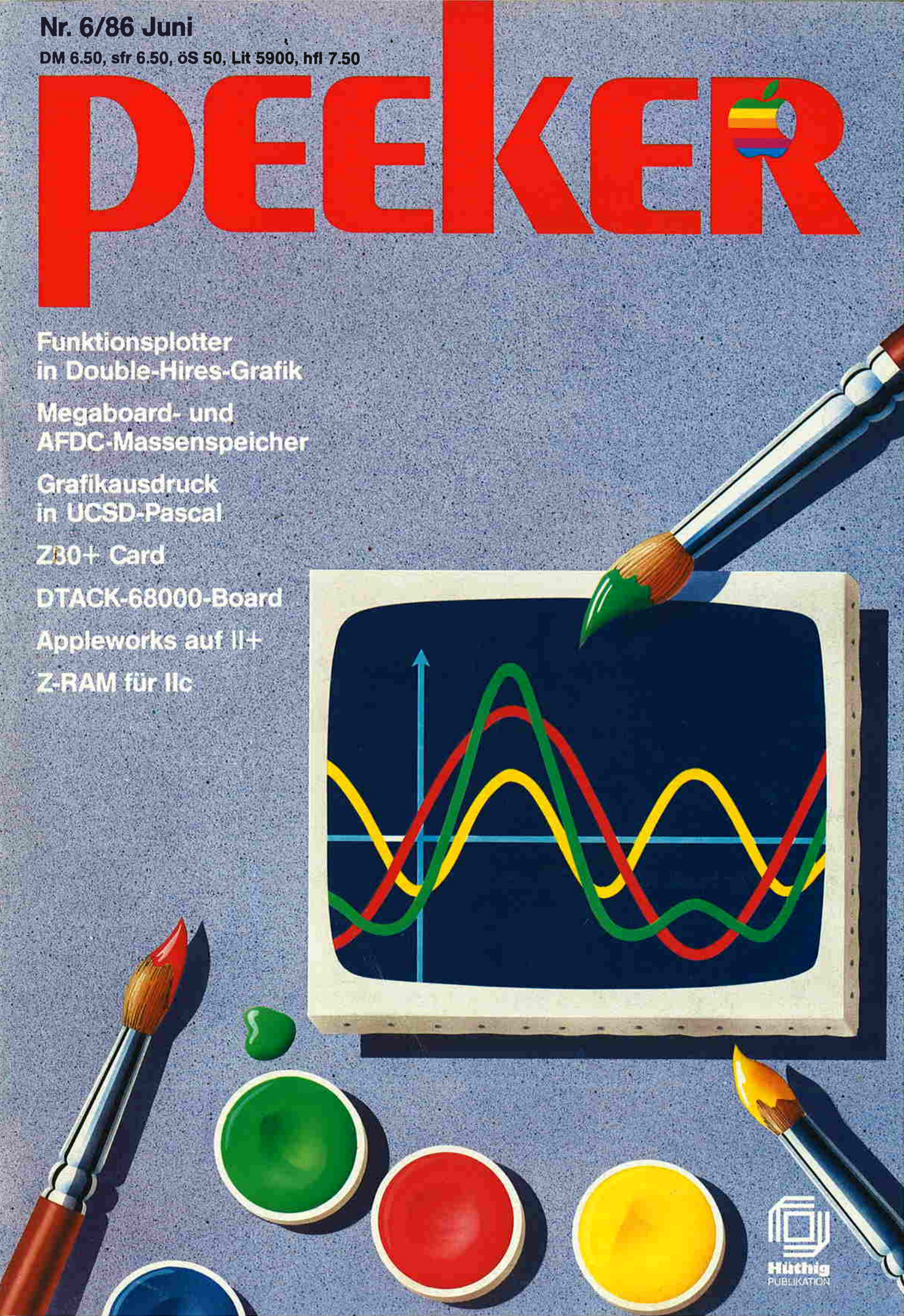
Grafikausdruck in UCSD-Pascal

Z80+ Card

DTACK-68000-Board

Appleworks auf II+

Z-RAM für IIc



Hüthig PUBLIKATION

Wichtige Information für unsere PEEKER-Leser

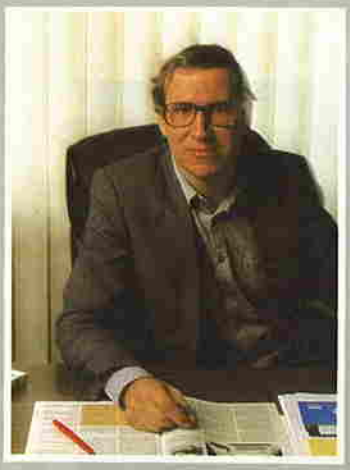
Sie erhalten Ihren »**peeker**«

- bei Ihrem **Zeitschriftenhändler am Bahnhof**
- bei allen **Apple-Händlern**
- und natürlich **beim Verlag.**

Denn der sicherste Weg, keine Ausgabe des »**peeker**« zu versäumen, ist noch immer das Abonnement zum Jahresvorzugspreis. Dabei sparen Sie bares Geld. Beachten Sie in diesem Zusammenhang unsere Anzeige auf Seite 9, oder rufen Sie uns an:
Telefon 0 62 21/489-281 (Peeker-Leserservice).

Hüthig PUBLIKATION

Dr. Alfred Hüthig Verlag, Im Weiher 10, 6900 Heidelberg
BTX ✱51851#



editorial

Kyan-Pascal Version 2.01A

Während der Drucklegung dieser Peeker-Ausgabe ist Kyan-Pascal in der Version 2.01A eingetroffen, welche die in diesem Heft beschriebenen Compiler-Fehler behebt. Die Änderungen berühren nur die Dateien PC, LIB und STDLIB.S, die zusätzlich auf die Kyan-Club-Disk #A aufgenommen wurden, die verschiedene Übungsprogramme sowie unsere FID-Library enthält (Preis DM 20,–; siehe Club-Rundschreiben #2 vom 15.5.86). Wenn Sie die Club-Disk #A nicht kaufen wollen, können Sie die geänderten 2.01A-Dateien kostenlos anfordern, falls Sie ein Diskettenduplikat der Disk 1, Seite 2 und eine Photokopie unserer Originalrechnung zur Version 2.0 als Kaufbeleg einschicken. Die neuen Dateien werden dann auf Ihr Duplikat überspielt. Kyan-Besteller, die ab 1.6.86 beliefert werden, erhalten automatisch die neue Version.

Das Adipositas-Syndrom

Im Vorfeld der Einführung des Apple-II-Nachfolgers, der Mitte September das Licht der Welt erblicken soll, möchte ich einem Problem nachgehen, das mir seit geraumer Zeit Sorge bereitet, nämlich dem paradoxen Trend in Richtung mehr Speicherkapazität und mehr Prozessorleistung bei weniger Mikro-Leistung.

Zunächst zur Speicherkapazität: 128K wären für einen 16-Bit-Mikrocomputer wohl etwas zu mager. Folglich werden Maschinen mit 512K und mehr angeboten. Doch hat man damit genügend *freien* Speicher? Keineswegs! Denn die Programme der Software-Häuser entwickeln sich immer mehr zu aufgedunsenen Speicherfressern. Das alte Visicalc beanspruchte für sich nicht viel mehr als 16K. Der Rest des Speichers stand für das Betriebssystem und die Tabellendaten zur Verfügung. Programme wie Lotus 1-2-3 usw. nehmen regelmäßig 200K bis 400K in Beschlag, so daß unter Abzug des Betriebssystems kaum noch Platz für die eigentlichen Daten bleibt. Beispielsweise belegt Framework II, „die Offenbarung für PC-Anwendungen" (so „Computer Persönlich", 12/86, S. 115), für das reine Programm 380K. Für mich ist dies keine Offenbarung, sondern eher eine Apokalypse. Ähnliches erleben wir beim Atari: Beim 512K-BASIC meldet der Fre-Befehl ca. 30K, also weniger als beim 48K-Applesoft. Mit dieser Fettsucht, die man als „Micro-Adipositas" bezeichnen könnte, geht eine Behäbigkeit in der Programmausführung einher, die bei größeren Datenmassen schlechthin zur Farce wird. Wenn man beim Wordstar 3.4 auf dem IBM-PC-XT 1000 A's eingibt und diese dann über „Ctrl-QA" in 1000 B's verwandeln will, so muß man exakt 309 Sekunden warten, also länger als beim alten Wordstar 3.0 unter CP/M mit 6MHz-Z80. Zum Vergleich ist der Fastwriter mit Accelerator in 0,125 Sekunden fertig und damit knapp 3000mal schneller als der Wordstar 3.4.

Trotzdem gilt der Wordstar 3.4 noch als „schnell" im Vergleich zum Wordstar 2000, bei dem man sich bis zu 15 Minuten gedulden muß, bis ein Block verschoben worden ist. Und der Sprung zum Ende eines sehr langen Textes dauert beim Wordstar 2000 4-5 Minuten gegenüber „nur" ca. 40-50 Sekunden beim Wordstar 3.4.

Nun könnte man mit Recht einwenden, daß diese Geschwindigkeitstests praxisfremd sind, denn welcher Anwender ersetzt schon 1000 A's durch 1000 B's. Tatsächlich sind die A's selbst unwichtig, doch was passiert, wenn nicht nach A's, sondern nach Adressen gesucht werden soll? Zu diesem Zweck habe ich – ebenfalls auf einem IBM-PC-XT – eine dBase-Testdatei mit 3000 Datensätzen mit einem 15stelligen Zufallsstring-Feld (Typ C) erzeugt und im Anschluß daran auf der Festplatte unter dBase III indiziert, was eine schiere Ewigkeit dauerte. Doch es kam noch schlimmer: Der LIST-Befehl, etwa LIST FOR FELD = „Z" – benötigte jetzt regelmäßig über 3 Minuten, gleichviel ob dBase III einen Record findet oder nicht.

Zurück zur praktischen Anwendung: Hier umfaßt ein Datensatz üblicherweise mehr als 1 Feld, und die Felder selbst sind meist mehr als 15 Zeichen lang. Außerdem benötigt man in der Regel mehr als eine Indexdatei. Zudem könnte man bei einer 10M-Festplatte auf die Idee kommen, eine Adreßdatei mit 10.000 Datensätzen anzulegen. Dann steigen jedoch die Verarbeitungszeiten von dBase III exponentiell an. Wartezeiten von weit mehr als einer Minute für das Einfügen von einer einzigen Adresse werden nunmehr zur Regel, und den LIST-FOR-Befehl kann man nur noch vor Geschäftsschluß eingeben: Am nächsten Morgen kommt man dann gerade rechtzeitig zur Anzeige der gefundenen Adressen ins Büro.

Woher kommt es nun, daß Wordstar 3.0, dBase III, Easy usw. gigantisch aufgedunsen und zugleich haarsträubend langsam sind? Liegt es am IBM-PC oder am MS-DOS? Mitnichten! Es liegt einzig und allein daran, daß diese Programme in höheren Programmiersprachen erstellt worden sind. So wurden etwa dBase III und Wordstar 3.0 in C geschrieben, und Easy – das neueste „fat-and-slow"-Produkt aus dem Hause Micropro – sogar in Modula 2. Nichts gegen C und Modula 2, aber der von höheren Sprachen erzeugte Objektcode stellt nun einmal keinen Ersatz für kompakte und schnelle Programme dar, die direkt in Assembler geschrieben wurden. Und wenn dann noch von Micropro behauptet wird, daß in den Wordstar 2000 fünfzig Mannjahre gesteckt wurden (s. „Computer-Magazin", 4/86, S. 44), dann muß man sich angesichts des dürftigen Resultats fragen, welche Amateure hier am Werk gewesen sind, denn selbst aus einer Compilersprache kann man einiges herausholen, wenn man intelligente Algorithmen zur Anwendung bringt.

Ich wünsche mir für Sie, liebe Leser, daß sich die Software-Häuser beim Apple-II-Nachfolger diesem unerquicklichen Trend nicht anschließen.

Ulrich Stiehl

inhalt

Impressum
Peeker
3. Jahrgang 1986
ISSN 0176–9200
© für den gesamten Inhalt einschließlich der Programme Dr. Alfred Hüthig Verlag, Heidelberg 1986
Verleger und Herausgeber:
Dipl.-Kfm. Holger Hüthig
Geschäftsführung Zeitschriften:
Heinz Melcher
Chefredakteur: Ulrich Stiehl (us)

Telefonnummern:
Zentrale: 06221/489-1
Redaktion: 06221/489-352
Anzeigen: 06221/489-206
Abonnement: 06221/489-283
Software: 06221/489-231
Bücher: 06221/489-353
(Bestellungen bitte nur schriftlich)

Abonnement:
Der Abonnent kann seine Bestellung innerhalb von 7 Tagen schriftlich durch Mitteilung an den Dr. Alfred Hüthig Verlag GmbH, Postfach 102869, 6900 Heidelberg 1, widerrufen. Zur Fristwahrung genügt die rechtzeitige Absendung des Widerrufs (Datum des Poststempels). Das Abonnement verlängert sich zu den jeweils gültigen Bedingungen um ein Jahr, wenn es nicht zwei Monate vor Jahresende schriftlich gekündigt wird. Die Abonnementsgelder werden jährlich im voraus in Rechnung gestellt, wobei bei Teilnahme am Lastschriftabbuchungsverfahren über die Postscheckämter und Bankinstitute eine vierteljährliche Abbuchung möglich ist.
Nichterscheinen infolge höherer Gewalt berechtigt nicht zu Ansprüchen gegen den Verlag.

Hüthig
PUBLIKATION

peeker

Heft 6/1986

Grafik

PLOT 3.E
Ein neuer Funktionsplotter
für den Apple IIe/c mit Double-Hires-Grafik
von Hans-Martin Eng 6

ProDOS

EDIT
Übungen zum Disk-Editor 24

Megaboard- und AFDC-Massenspeicher
Mit ProDOS-Backup-Programmen
für die MDB-Festplatte
und das Erphi-Subsystem
von Ulrich Stiehl 30

UCSD

SUPERDUMP für Apple Pascal 1.1 und 1.2
Grafik-Ausdruck für Epson-FX80
und Imagewriter I
von Jürgen Geiß 40

Kyan

Kyan-Pascal 2.0
Tips und Tricks
von Ulrich Stiehl 47

CP/M

Adreßverwaltung mit dBase II
Listing des überarbeiteten Quelltextes
von Ing. (grad.) Ernst Fischer 52

Produkte

Erfahrungs- und Testberichte

DTACK-68000-Board
Ein Erfahrungsbericht
von Wolfgang Trier 58

Plusworks
Appleworks auf dem Apple II+
Ein Erfahrungsbericht
von Herwig Cuypers 60

Z-RAM für den Apple IIc
Ein Erfahrungsbericht
von Ulrich Tönnies 62

Sidekick
Macintosh-Büro-Manager 66

Z80+ Card
getestet von Hans-Martin Eng 67

Leserbriefe

Jörg Bliesener beantwortet
CP/M- und Wordstar-Leserbriefe 68

Kurzberichte

Analytisches Inhaltsverzeichnis
der Peeker-Sammeldisketten #1 bis #18 64

Anschrift:
Dr. Alfred Hüthig Verlag GmbH
Im Weiher 10, Postfach 10 28 69
6900 Heidelberg
Telefon (0 62 21) 4 89-1
Telex 4-6 17 27 hued d.
Telefax (0 62 21) 489 279
BTX * 51851 #

Vertrieb:
Erscheinungsweise: 12 Hefte jährlich,
Erscheinungstag jeweils 1 Woche vor Monatsbeginn.
Jahresabonnement Inland DM 72,–, einschl. MwSt und Versandkosten.
Jahresabonnement Ausland DM 72,– plus DM 18,– Versandkosten.
Einzelheft DM 6,50
Vertrieb Handel:
MZV – Moderner Zeitschriften Vertrieb GmbH
Breslauer Str. 5, Postfach 1123,
8057 Eching b. München,
Tel. 089/31 90 06 13, Telex 0 522 656
Vertriebsleitung:
Walter Menzel, Tel. (0 62 21) 48 92 80

Bankverbindungen:
Zahlungen: an den Dr. Alfred Hüthig Verlag GmbH, D-6900 Heidelberg 1: Postgirokonten: BRD: Ludwigshafen 4799-673, BLZ 545 100 67; Österreich: Wien 75558 88; Schweiz: Basel 40-24417; Niederlande: Den Haag 1 457 28; Italien: Mailand 5 968 92 08; Belgien: Brüssel 1084 1261; Dänemark: Kopenhagen 603 4969; Norwegen: Oslo 199 4243; Schweden: Stockholm 5477 76-5
Bankkonten: Landeszentralbank Heidelberg 67 207 341; BLZ 672 000 00; Deutsche Bank Heidelberg 02 65 041; BLZ 672 700 03; Bezirkssparkasse Heidelberg 204 51, BLZ 672 500 20.

Herstellung:
Produktionsleitung: Gunter Sokollek
Gestaltung: Rainer Schmitt
Titelbild: Werner Hable
Satz und Druck:
Heidelberger Verlagsanstalt
Printed in Germany

„Peeker" ist eine unabhängige Zeitschrift. Sie ist nicht verbunden mit der Firma Apple Computer, Inc. oder der Apple Computer GmbH. APPLE, das Apple-Zeichen und MAC sind Warenzeichen der Firma Apple Computer, Inc. und MACINTOSH ist ein Warenzeichen, in Lizenz vergeben von der Firma McIntosh Laboratory an die Firma Apple Computer, Inc.

PLOT 3.E

Ein neuer Funktionsplotter für den Apple IIe/c mit Double-Hires-Grafik

von Hans-Martin Eng

Vielleicht erinnern Sie sich noch an das Programm PLOT.2.0, das im Peeker, Heft 5/85 veröffentlicht wurde. Der Nachfolger PLOT.3.E bietet doppelte Grafikauflösung sowie ausführlichere Erläuterungen und komfortablere Bedienung.

PLOT.3.E ist das direkte Nachfolgeprogramm meines im Mai 1985 veröffentlichten PLOT.2.0. Dieses wurde auf einem Apple II+-kompatiblen Rechner, ausgerüstet mit einem speziellen Druckerinterface, entwickelt und fiel daher extrem gerätespezifisch aus. Herr Stiehl bat mich schon vor einiger Zeit, dieses Programm so umzuschreiben, daß es doppelt hochauflösende Grafik erstellt und auf dem Imagewriter ausdruckt. Die doppelt hochauflösende Grafik beherrschte PLOT.2.0 schon bald, aber ich sah mich außerstande, es an den Imagewriter anzupassen. Um ein Programm auf einem speziellen Drucker zum Laufen zu bringen, muß man diesen schon zur Verfügung haben. Außerdem stellte sich schon bald heraus, daß das Zeichnen doch sehr langsam vor sich ging.

Darum entschloß ich mich, eine neue, verbesserte Version zu erstellen, die nun mehr über folgende Erweiterungen verfügt:

1. Schnelles Zeichnen doppelt hochauflösender Grafik,
2. höherer Benutzerkomfort,
3. weitgehende Hardware-Unabhängigkeit.

Dieser Artikel gliedert sich in sieben Abschnitte:

1. Erläuterungen zur Minimalkonfiguration und Anweisungen zur Installation von PLOT.3.E unter DOS 3.3
2. Ausführliche Bedienungsanleitung anhand von Beispielen

3. Kurze technische Erläuterungen
3.1. zum Applesoft-Programm
3.2. zu den Assemblerroutinen
4. Anweisungen zum Ausdrucken der Grafiken mit SUPERDUMP
5. Erläuterungen zur Anpassung von PLOT.3.E an individuelle Gerätekonfigurationen

1. Minimalkonfiguration und Installation

Als Minimalkonfiguration benötigen Sie entweder einen Apple IIe mit erweiterter 80-Zeichenkarte oder einen Apple IIc. Als Option können Sie mit einem grafikfähigen Matrixdrucker, für den Sie SUPERDUMP angepaßt haben, die von PLOT.3.E erstellten Grafiken ausdrucken. Lesen Sie dazu bitte Abschnitt 4 und 5. PLOT.3.E ist auf die Benutzung unter DOS 3.3 (oder einem verwandten Betriebssystem) zugeschnitten, ist jedoch auch unter ProDOS lauffähig.

Zur Installation von PLOT.3.E unter DOS 3.3 gehen Sie bitte wie folgt vor:
1. Booten Sie DOS 3.3 (oder besser Diversi-DOS)
2. Geben Sie ein:

```
NEW
10 NEW
POKE 40514,20
```

3. Legen Sie nun eine Leerdiskette ein und initialisieren Sie sie mit
INIT PLOT.3.PRO.
Wenn diese Diskette gebootet wird, erwartet DOS 3.3 einen EXEC-File als HELLO-Programm.
4. Löschen Sie den File wieder mit
DELETE PLOT.3.PRO.
5. Transferieren Sie nun mit FID oder einem anderen File-Kopierprogramm folgende Files auf die Leerdiskette.

```
PLOT.3.PRO
PLOT.3.E
PLOT.BX
PLOT.HELP.1
PLOT.HELP.2
PLOT.HELP.3
PLOT.HELP.4
PLOT.HELP.5
```

Diese Files (und noch mehr) befinden sich auf der Sammeldiskette. Ich würde Ihnen nicht empfehlen, all diese Files selbst einzutippen, es sei denn, Sie können die Arbeit auf mehrere Freunde verteilen. Wollen Sie es dennoch wagen, so beachten Sie bitte die Hinweise vor den Listings. Haben Sie dies alles bewerkstelligt, läuft PLOT.3.E automatisch an, wenn Sie die Diskette booten.

2. Bedienungsanleitung

Diese Anleitung ist wenig sinnvoll, wenn Sie die Beispiele nicht selbst ausprobieren. Setzen Sie sich also an Ihren Rechner und werfen Sie PLOT.3.E an, am besten durch Booten einer Diskette, die wie oben beschrieben initialisiert wurde. Ansonsten mit *EXEC PLOT.3.PRO.*

Nach kurzer Zeit erscheint auf dem Bildschirm ein kurzes Startmenü. Der Rechner wartet nun auf einen Tastendruck. Danach erscheint auf Ihrem Monitor die Frage „Wünschen Sie eine Kurzeinführung? Betätigen Sie bitte 'J' oder 'N'." Eine ähnliche Frage wird PLOT.3.E beim Eintritt in jeden größeren Programmteil stellen. Fragen, die Sie mit „J" oder „N" beantworten müssen, erkennen sie durch ein angehängtes „<J/N>". Tippen Sie nun auf *J*, so werden die Erläuterungen zum betreffenden Programmteil von der Diskette auf dem Bildschirm ausgegeben. Da die Bedienungsanleitung, die Sie jetzt gerade lesen, wesentlich ausführlicher ist, können Sie auf *N* tippen und somit auf die Erläuterungen verzichten.

Danach erscheint auf dem Bildschirm die Frage: „Funktionseingabe: Wünschen Sie Erläuterungen? <J/N>". Wählen Sie! Später werden Sie aufgefordert, die Funktion einzugeben, deren grafische Darstellung Sie wünschen. Als erstes Beispiel tippen Sie bitte *SIN(X)* ein, gefolgt von einem Druck auf die Return-Taste. Haben Sie dabei keinen Fehler gemacht, erscheint die Frage „Ist das richtig? <J/N>". Tippen Sie auf *N*, so können Sie Ihre Eingabe korrigieren. Ist Ihnen jedoch ein gravierender Fehler unterlaufen (z.B. eine nicht geschlossene Klammer), ertönt ein Piepston und PLOT.3.E fordert Sie auf, den Fehler zu korrigieren.

Als nächstes fragt PLOT.3.E, ob Sie Erläuterungen zur Eingabe der Darstellungsgrenzen abrufen wollen. Ist das erledigt, geben Sie bitte nacheinander, erst für die X-, dann für die Y-Achse, deren Anfangs- und Endwert ein. In unserem Beispiel wählen wir *0, 6.28, -1.1*, und *+1.1*. Tippen Sie das genauso ein, gefolgt von einem *Return*. Unterläuft Ihnen bei der Eingabe ein Syntax-Fehler, erfolgt eine Fehlermeldung, und Sie werden gebeten, die Eingabe zu wiederholen. Ist die Eingabe erfolgreich abgeschlossen, werden auf der unteren Hälfte des Bildschirms die festgelegten Darstellungsgrenzen gezeigt. Ist alles nach Wunsch eingegeben, tippen Sie bitte auf *J*, tippen Sie jedoch auf *N*, können Sie die Eingabe wiederholen. (Wählen Sie vorerst die Option der vollständigen Neueingabe. Die zweite Option führe ich Ihnen später vor.)

ProDOS-Editor 1.0

Applesoft-Editor unter ProDOS-Betriebssystem

von U. Stiehl

1984, Diskette und Manual, DM 98,–
ISBN 3-7785-1024-X

Mit diesem neuen Editor – übrigens der bislang einzige deutsche ProDOS-Editor – wird dem Applesoft-Programmierer ein Werkzeug zur effektiven Programmierung unter dem Betriebssystem ProDOS gegeben, denn die früheren Editoren sind allesamt unter ProDOS nicht mehr lauffähig.

Unter anderem sind folgende Features implementiert worden:

- Zeilenorientierter Editor mit jedem erdenklichen Redigierkomfort (Insert, Delete, Tab, Restore, freie Cursorbewegung in allen vier Richtungen, Eingabe von Ctrl-Buchstaben in Applesoft-Zeilen usw.)
- Renumber (Zeilen-Umnumerierung)
- Xreference (sortierte Variablenliste)
- Suchen von Tokens, Strings und Variablen
- dezimale und hexadezimale Umrechnungen
- Ausführung von Monitorbefehlen aus dem Editor heraus
- Listen des Applesoft-Programms in speicherinterner Form als Hex-Dump
- Suchen von Hex-Folgen, Adressen oder Speicherstellen im gesamten RAM-Bereich einschließlich der Language-Card
- frei definierbare Tastatur-Macrobefehle

Der Applesoft-Editor liegt in einem von ProDOS geschützten Bereich und läßt sich per Tastendruck vorübergehend abschalten und ebenso einfach wieder aktivieren.

Gerätevoraussetzung: Apple II+, IIe oder IIc, 40 Zeichen/Zeile

Hinweis: Der Applesoft-Editor für DOS 3.3 befindet sich auf Sammeldisk # 16 für Fortsetzungsbezieher. Siehe Peeker 4/86, S. 13.

Hüthig Software Service,
Postfach 10 28 69,
D-6900 Heidelberg

Auf dem Bildschirm müßte nun folgende Zusammenfassung erscheinen:

„Funktion: Y=SIN(X)
Parameter: Es werden keine Parameter verwendet.
Darstellungsgrenzen: X: 0 .. 6.28, Y: -1.1 .. 1.1"

Wieder folgt die Frage, ob alles richtig ist.

Wenn Sie nun auf *J* tippen, wird der Bildschirm gelöscht, ein Achsenkreuz gezeichnet und eine Sinuslinie langsam von links nach rechts durchgezogen. Das dauert insgesamt etwa 80 Sekunden. Tippen Sie während des Zeichnens auf die Leertaste, sehen Sie statt der Grafik Text. In diesem Text sind Ihre Eingabedaten aufgeführt. Tippen Sie nochmals auf die Leertaste, erscheint wieder die Grafik.
Ist die Sinuslinie bis zum rechten Bildschirmrand durchgezogen, ertönt ein Piepston. Sie sollte nun so aussehen, wie **Bild 1**. Drücken Sie nun auf irgendeine Taste, so tritt PLOT.3.E ins Hauptmenü ein. Von dort aus können Sie alle zentralen Funktionen von PLOT.3.E aufrufen. Dieses Hauptmenü erscheint auch, wenn Sie während eines Zeichenvorgangs auf *Ctrl-C* gedrückt haben.
Ich möchte Ihnen nun kurz die einzelnen Funktionen des Hauptmenüs vorführen. Tippen Sie auf *1*! Richtig, Sie können die Darstellungsgrenzen erneut eingeben. Beantworten Sie also die Frage „Ist das richtig?" mit *N*! Geben Sie folgende Werte ein: X: *-3.14*, *3.14* und Y: *0*, *0*. Ergebnis: Anfangs- und Endwert für die X-Achse werden korrekt übernommen. Für die Y-Achse jedoch erscheinen die Werte +/-2.1496... . Tippen Sie bitte auf *J*. Die Grafik wird gelöscht und ein neues Achsenkreuz gezeichnet. Danach erscheint wieder das Hauptmenü.
Drücken Sie auf die Leertaste, damit Sie sich das Achsenkreuz genauer ansehen können. Sie werden feststellen, daß nun die Markierungen auf der X-Achse untereinander den gleichen Abstand haben wie die auf der Y-Achse. Das heißt, daß X- und Y-Achse auf den gleichen Maßstab abbilden. Probieren Sie es aus. Tippen Sie auf *6*, dann wird die Funktion neu gezeichnet. Richtig, in den Nullstellen hat die Sinuslinie nun die Steigung +/-1, wie wir es in der Schule gelernt haben.
Fassen wir zusammen. Wenn Sie für *eine* Achse korrekte Anfangs- und Endwerte eingeben, für die *andere* Achse beide Male *0*, dann wird der Maßstab dieser Achse gleich dem der anderen. Die Darstellungsgrenzen dieser Achse sind gleich weit vom Ursprung entfernt.
Probieren wir's gleich nochmals aus. Geben Sie nun für X: *+1*, *+6.28* ein, für Y wieder beide Male *0*. Was ist nun passiert? Die Darstellungsgrenzen sind unverändert geblieben. Begründung: Da für die X-Achse Anfangs- und Endwert größer als 0 sind, könnte die Y-Achse nicht gezeichnet werden (da diese durch den Nullpunkt der X-Achse geht). Darum korrigiert PLOT.3.E notfalls Ihre Eingaben so, daß alle Achsen gerade noch gezeichnet werden können. In unserem Beispiel wird daher der Anfangswert für X auf 0 festgesetzt.

Tippen Sie nun nochmals auf *N* für nicht richtig. Wir wollen jetzt die zweite Möglichkeit für die Festlegung der Darstellungsgrenzen kennenlernen. Tippen Sie also gleich nochmal auf *N*. Offensichtlich tritt PLOT.3.E in einen neuen Programmteil ein, denn es bietet Ihnen wieder Erläuterungen an. Was können Sie nun in diesem Programmteil tun?

a) Sie können die X- und die Y-Achse getrennt stauchen oder strecken. Dazu fragt Sie der Rechner für jede Achse, ob Sie sie manipulieren möchten. Tippen Sie auf *J*, bittet er sie, das Verhältnis der alten Darstellungsgrenzen zu den neuen einzugeben. Es seien die alten Grenzen -2 und +4. Geben Sie als Verhältnis
alte Grenzen: *2*,
neue Grenzen: *3*
ein, werden die Grenzen auf -3 und +6 festgelegt. Verstanden? Geben Sie die Zahlen getrennt ein, jeweils von *Return* abgeschlossen.

b) Nachdem Sie gefragt worden sind, ob Sie die Achsen strecken oder stauchen wollen, dies eventuell auch getan haben, können Sie nun den Nullpunkt auf jeder Achse verschieben. Dies geschieht, indem Sie wieder ein Zahlenverhältnis eingeben, und zwar das des positiven zum negativen Halbast der betreffenden Achse. Seien also die Grenzen wieder -2 und +4. Geben Sie als Verhältnis *1:0* an, werden die neuen Darstellungsgrenzen auf 0 und +6 festgelegt. Würden Sie als Verhältnis *1:1* eingeben, wären die neuen Grenzen -3 und +3.

In unserem konkreten Fall *strecken* Sie bitte die Y-Achse um den *Faktor 2*, das heißt, *Verhältnis alt:neu = 2:1*. Dann wollen wir noch für die X-Achse das Verhältnis der positiven und negativen Halbachse auf *1:1* festlegen. Plotten Sie die Funktion mit Option *6* des Hauptmenüs. Das Ergebnis müßte aussehen wie **Bild 2**.

Nun wollen wir eine neue Funktion eingeben: Wählen Sie dazu im Hauptmenü Option *3*. Geben Sie als Funktion *SQR(R*R-X*X)* ein. Dies ist eine Halbkreisfunktion mit dem Kreisradius R. Dieses R stellt einen sogenannten Scharparameter dar (*Schar*parameter deshalb, weil so statt einer Funktion eine Funktionenschar dargestellt wird.) Ist die Eingabe abgeschlossen, bittet Sie PLOT.3.E gleich, R mit einem Wert zu belegen. Geben Sie *5* ein.
Als nächstes müssen wir die Darstellungsgrenzen neu eingeben. Geben Sie bitte zuerst für X: *-5* und *+5* ein, für Y: *0* und *0*. Verschieben Sie dann gleich den Nullpunkt auf der Y-Achse an den unteren Bildschirmrand (Verhältnis positive : negative Halbachse = *1:0*). Lassen Sie diese Funktion plotten. Es kommt dann tatsächlich ein Halbkreis heraus.
Wählen Sie jetzt Option *2* des Hauptmenüs. Geben Sie als neuen Wert für R nun *4* ein und lassen Sie die Funktion wieder plotten. Der Halbkreis beginnt erst bei X=−4 und endet bei X=+4. Der Grund dafür ist, daß an Stellen, an denen der Betrag von X größer als 4 ist, das Argument der Wurzel kleiner als 0 wird. Solche Funktionswerte sind jedoch (zumindest im Körper der reellen Zahlen) nicht definiert. Normalerweise würde Applesoft das Programm anhalten und eine Fehlermeldung ausgeben. Diese fängt PLOT.3.E jedoch intern ab, so daß Sie sich nicht allzusehr um den Definitionsbereich der Funktion kümmern müssen.

Rufen Sie bitte nochmals die Funktionseingabe auf. Ändern Sie innerhalb der Klammern *R* in *T* um. Nach erfolgter Eingabe fragt PLOT.3.E, ob Sie T als Variable einer Schleife mit fester Schrittweite verwenden wollen. Tippen Sie auf *J*. Danach geben Sie als Anfangswert für T *5*, als Endwert *1* und als Schrittweite *-1* ein. Löschen Sie mit Option *4* den Bildschirm und lassen Sie mit Option *6* die Funktion wieder zeichnen. Sie können feststellen, daß, nachdem ein Halbkreis fertig gezeichnet worden ist, nicht mehr der Piepston ertönt, sondern ein neuer Halbkreis weiter innen gezeichnet wird, bis zum Schluß der Halbkreis mit Radius 1 fertig ist. Dann erst erfolgt der Piepston. Sie sehen fünf konzentrische Halbkreise auf dem Bildschirm, wie in **Bild 3**.
Rufen Sie nun Option *5* des Hauptmenüs auf. Sie sehen die Datenliste, die normalerweise während des Zeichnens auf dem Textbildschirm steht, komplett mit allen Parameterwerten, inklusive einem, der noch gar nicht aktuell ist.
Nun müssen Sie nur noch die Optionen *7* und *0* des Hauptmenüs kennenlernen. *0* erlaubt Ihnen den sanften Ausstieg aus PLOT.3.E. Brutal geht's auch mit Ctrl-Reset. Option *7* ruft ein neues Menü auf, nämlich das **Disk-Menü**. Wie der Name andeutet, können von dort aus Diskettenoperationen ausgeführt werden. Das wären:

Apple II Raster Grafik Stanton

Das Buch führt in die Eigenschaften der hochauflösenden Apple-Grafik ein und präsentiert eine Reihe schneller Assembler-Routinen, mit denen Sie viele Effekte bekannter Arcade-Spiele selbst programmieren können.
299 Seiten, komplett mit Disk
3-89058-008-4 DM 49,-

Apple II Schaltpläne Gayler

Wenn Sie Ihren Apple II oder IIplus selbst reparieren, Interfaces entwerfen oder einfach nur besser über das Innenleben Ihres Computers Bescheid wissen wollen: Informationen, Schaltpläne und Diagramme, Theorie und praktische Tips.
215 Seiten, DIN A4
3-89058-012-2 DM 64,-

Apple II Assembler Programmierung Wagner

Das Assembler-Lehrbuch für BASIC-Kenner!
Eine stufenweise Einführung in die Befehle und Strukturen der 6502-Assemblersprache, mit vielen Beispielen von der einfachen Tonerzeugung bis zum Diskettenzugriff.
277 Seiten, komplett mit Disk
3-89058-005-X DM 49,-

Ampersand Verlag

Teltower Damm 168
D-1000 Berlin 37
(030) 815 80 69

Begleitdisketten jetzt gratis!

Für den gleichen Inhalt sollen Sie nicht zweimal zahlen müssen! Die Begleitdiskette ist ab sofort im Preis des Buches enthalten!

Preissenkung!

Visible Computer

Ein Simulationsprogramm, das Sie in das Innere des 6502-Mikroprozessors führt.
Sie sehen auf dem Bildschirm, wie die einzelnen Instruktionen in Zeitlupe ausgeführt werden, wie sich Register und Flags verändern. Ein ideales Hilfsmittel zum Erlernen der Assemblerprogrammierung, danach ein wertvolles Werkzeug beim Testen Ihrer eigenen Programme. Komplett mit einem 6502 Editor/Assembler und einem Lehrbuch zur Maschinenprogrammierung (deutsch, 150 Seiten).
3-89058-019-X DM 58,-

NEU!

MERLIN PRO

Info anfordern!

Ein professioneller Macro-Assembler für Apple IIe und IIc.
Jetzt neu:
- Relocating Linker zum automatischen Linken unabhängiger, verschiebbarer Module
- 6502, 65C02 und 65802
- DOS 3.3 und ProDOS

dtsch. Handb., benötigt 128K
3-89058-026-2 DM 298,-

Weiterhin lieferbar:

MERLIN

wie MERLIN PRO, aber ohne Linker, 65802 und ProDOS. läuft auf jedem Apple II mit mind. 64K!
mit deutschem Handbuch
3-89058-024-6 DM 198,-

Apple Pascal Grafik Swan

Programme, mit denen Sie die Grafikmöglichkeiten des Apple voll ausschöpfen. DESIGNER: Entwurf eigener Zeichensätze. GREDIT: komplexer Grafikeditor. PRINTFOTO: Grafikdump auf den Drucker. Viele fertige Prozeduren zum Einbauen.
280 Seiten, komplett mit Disk
3-89058-011-4 DM 49,-

Applesoft Trickkiste Golding

INPUT-ANYTHING, PRINT USING, schnelle Garbage Collection und viele andere nützliche Utilities, dazu detaillierte Informationen über die Arbeitsweise des BASIC-Interpreters, Einsprungadressen, Systemvariablen.
304 Seiten, komplett mit Disk
3-89058-035-1 DM 44,-

Rückgaberecht!

Wenn Ihnen ein Titel wider Erwarten nicht gefällt, dann senden Sie ihn innerhalb von 10 Tagen zurück und bekommen den Kaufpreis erstattet!
Gilt für Bücher und Software!

Apple ProDOS Handbuch Worth

Worth/Lechner sind die Autoren von "Beneath Apple DOS", der Bibel aller DOS 3.3-Benutzer. In Ihrem neuen Buch haben sie sich ProDOS vorgenommen: Arbeitsweise, MLI, BASIC.SYSTEM. Das Standardwerk für jeden ProDOS-User.
270 Seiten, komlett mit Disk
3-89058-038-6 DM 46,-

Apple Pascal Trickkiste DeGroat

Utilities, Informationen, Tips und Tricks:
P-Code-Decodierer, File-Konverter (von DOS nach Pascal und zurück), Disk-Zapper, Grafik-Routinen, Textformatierer, verbesserte Ein-/Ausgabe-Prozeduren u.a.m.
248 Seiten, komplett mit Disk
3-89058-032-7 DM 48,-

NEU!

Apple IIe Hardware Sather

Eine komplette Beschreibung der Schaltungen, RAM, ROM, I/O, Video, etc.. Unverzichtbar für den Entwickler und Techniker, interessant für jeden Apple IIe-Besitzer, der sich über die Funktionsweise seines Computers informieren will.
Ca. 350 Seiten, DIN A4
3-89058-040-8 DM 78,-

Name: .....
Anschrift: .....
.....

○ *V-scheck Liegt bei (Spesenfreie Lieferung)*
○ *per Nachnahme (zzgl. DM 3,– Versandspesen)*

Anz	Titel oder ISBN-Nummer	Preis

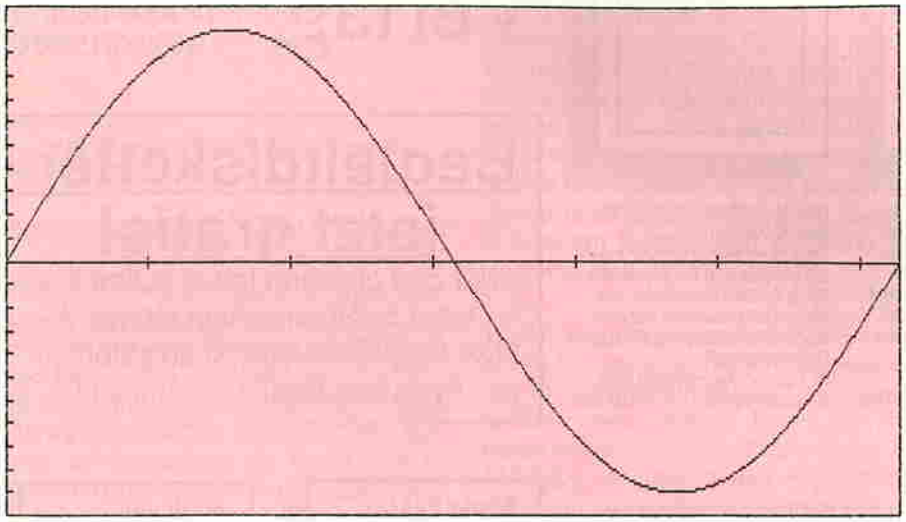


Bild 1

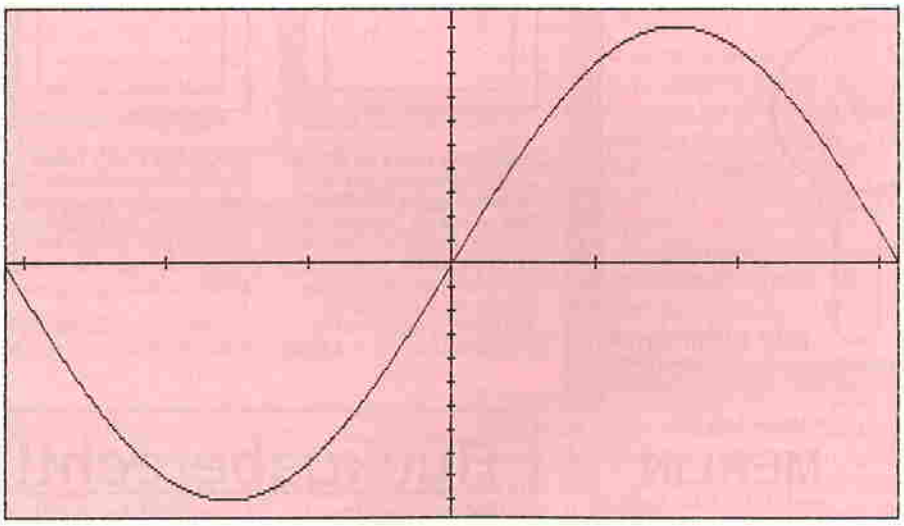


Bild 2

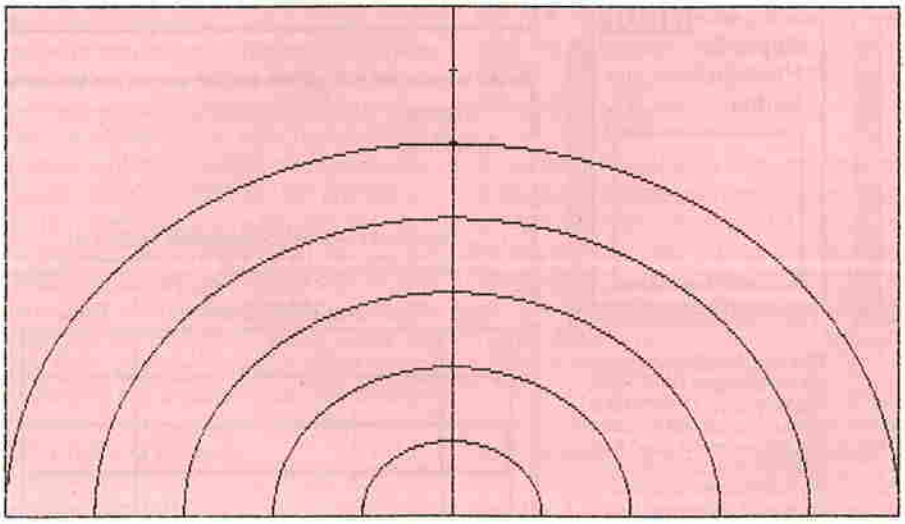


Bild 3

1. Zeigt den CATALOG des aktiven Diskettenlaufwerks an.
2. Lädt eine Grafik vom aktiven Laufwerk in den Speicher.
3. Schreibt die im Speicher befindliche Grafik in zwei Binärfiles auf der aktiven Diskette.
4. Wählt das aktive Laufwerk aus.
5. Ermöglicht Ihnen jeden gewünschten PLOT.HELP.-File nochmals zu lesen.
6. Startet, abhängig von der von Ihnen verwendeten Version, entweder das Programm SUPERDUMP aus Peeker, Heft 6/85 (von Sammeldisk #5) oder druckt die Grafik direkt aus. Lesen Sie bitte hierzu Abschnitt 5.
0. Bringt Sie schließlich wieder sicher ins Hauptmenü zurück.

Probieren wir zuerst *3* aus. PLOT.3.E bittet Sie, einen File-Sammelnamen einzugeben. *KREISE* wäre keine schlechte Idee. Kurze Zeit später erscheint wieder das Disk-Menü. Der CATALOG (Option 1) enthält nun zwei Binärfiles mehr, der eine heißt KREISE.A, der andere KREISE.M, beide sind je 34 Sektoren lang.
Löschen Sie nun die Grafik und wählen Sie Disk-Menü-Option *2*. Geben Sie als Sammelnamen wieder *KREISE* ein. Nach kurzer Zeit ist das alte Bild wieder zu sehen.
Option *4* zeigt zuerst die Slot-Drive-Kombination des aktiven Laufwerks an. Tippen Sie nun zuerst auf eine der Tasten *1 bis 7*, je nachdem, unter welcher Slotnummer das gewünschte Laufwerk zu finden ist. Dann tippen Sie bitte noch auf *1 oder 2*, je nachdem, welches Laufwerk Sie letztendlich wünschen. **Achtung!** PLOT.3.E prüft nicht nach, ob unter der aktuellen Slot-Drive-Kombination auch ein Laufwerk zu finden ist. Der entstehende I/O-Error ist jedoch nicht weiter tragisch.
Tippen Sie nun im Disk-Menü auf *0*, dann gleich *nochmals*, und Sie sind fertig. Sie haben alle Funktionen von PLOT.3.E kennengelernt.

3. Technische Erläuterungen

3.1. zum Applesoft-Programm

Das Applesoft-Programm ist mit einigen Ampersand-Befehlen (&) angereichert:

Zuerst die Grafik-Befehle:
& HPLOT x1,y1 (**TO** x2,y2) – ist syntaktisch wie der Applesoft-Befehl **HPLOT** aufgebaut, bietet auch die gleichen Möglichkeiten. Er arbeitet nur in doppelt hochauflösender Grafik.
& HGR – Schaltet Double-Hires-Grafik ein und löscht den Bildschirm.

& HGR2 – Schaltet Double-Hires-Grafik ein, ohne den Bildschirm zu löschen.
& TEXT – Schaltet von der Double-Hires-Grafik auf 40-Zeichen-Textdarstellung zurück.

Nun die restlichen &-Befehle:
& POS (x,y) ist ein Text-Befehl für die 80-Zeichendarstellung und positioniert den Cursor in Spalte x, Zeile y.

& FN, F$ – liest eine BASIC-Funktion von der Tastatur ein und übergibt diese einmal als **FN Y(X)** an das Hauptprogramm, andererseits als String an die Stringvariable F$.

Die grundlegende Arbeitsweise des Programms werde ich diesmal nicht erörtern, da diese Algorithmen im wesentlichen unverändert geblieben sind. Lediglich ihre Formulierung in Applesoft und vor allem die Reihenfolge im Programmtext haben sich geändert.

So gibt es nun einen einheitlichen Aufbau. Die Zeilen 1010 bis 4900 enthalten Subroutinen, dann schließt sich das Disk-Menü an, gefolgt vom Programmkern mit Initialisierungsteil ab Zeile 6000 und Hauptmenü ab Zeile 6460. Zum Schluß folgt noch der Programmteil, der aufgerufen wird, wenn PLOT.3.E versehentlich direkt durch *RUN PLOT.3.E* gestartet wurde, ohne den Pointer in der Zero-Page $0067 heraufzusetzen.

Eingehendere Erläuterungen unterlasse ich, da das Programm sich recht gut selbst dokumentiert und sich außerdem seit PLOT.2.0 die Algorithmen nicht grundlegend geändert haben.

3.2. Zum Assemblerprogramm

Sie haben sicherlich bemerkt, daß das Assemblerprogramm nur auszugsweise abgedruckt ist. Es umfaßt im Original nämlich 940 Zeilen, zum Abdrucken etwas zuviel. Daher bringen wir nur den Hexdump des Programms komplett, außerdem noch einen Auszug aus dem Assemblerlisting. Das komplette Listing ist auf der Sammeldiskette zu finden.

Genauer eingehen möchte ich daher auch nur auf den Algorithmus, der hinter dem veröffentlichten Auszug steckt. Es geht um den Algorithmus, mit dem man Linien zwischen zwei Punkten zieht. Dazu möchte ich etwas weiter ausholen:

3.2.1. Der normale Treppenalgorithmus

Stellen Sie sich einen Maschendraht vor, dazu ein Stück Bindfaden und spannen Sie in Gedanken den Bindfaden zwischen zwei Kreuzungspunkten. Frage: Wie oft trifft der Bindfaden genau einen Kreuzungspunkt? Im Allgemeinfall wird das nicht allzuoft passieren. Dieses Problem besteht auch, wenn Sie unter Rastergrafik zwei Punkte durch eine Gerade verbinden wollen. Man behilft sich heutzutage dadurch, daß man die Gerade durch eine Art Treppenlinie annähert. Das geht recht schnell und ist einfach in Maschinensprache zu formulieren. Auch das Applesoft-ROM geht so vor. Die Geraden sehen dann, vergrößert abgebildet, so aus, wie es das **Bild 5** darstellt.

3.2.2. Heuristische Herleitung des Algorithmus

Betrachten Sie bitte **Bild 4**. Von den beiden eingezeichneten Geraden interessiert uns momentan nur die untere, d.h. Gerade g_1. Sie geht vom Anfangspunkt A_1 mit den Koordinaten ($x_{A,1}$, $y_{A,1}$) zum Endpunkt B_1 ($x_{B,1}$, $y_{B,1}$). Hervorgehoben sind außerdem die Punkte P_1 und Q_1. Sie werden durch den Treppenalgorithmus gefunden.

Das zentrale Problem ist, ein Kriterium zu finden, das etwas darüber aussagt, ob ein Punkt oberhalb oder unterhalb der zu zeichnenden Gerade liegt. Betrachten wir nun die Punkte P_1, Q_1 und B_1.

B_1 liegt naturgemäß genau auf g_1. Bezeichnen wir mit n_x (P) die Anzahl der Schritte in x-Richtung bis zum Punkt P, analog n_y (P). Dann führen wir noch Δ_x und Δ_y als die Differenz der Koordinaten von Anfangs- und Endpunkt. Somit gilt:

$$(1)\quad \frac{n_y(B)}{n_x(B)} = \frac{\Delta Y}{\Delta X}$$

$$(2)\quad \frac{n_y(P)}{n_x(P)} < \frac{\Delta Y}{\Delta X}$$

$$(3)\quad \frac{n_y(Q)}{n_x(Q)} > \frac{\Delta Y}{\Delta X}$$

Es ist nicht einzusehen, warum wir nicht verallgemeinern dürfen, d.h. Gleichung (1) ist nur erfüllt, wenn der Punkt auf der Geraden liegt, und die Ungleichungen (2) und (3) gelten, wenn der Punkt unterhalb bzw. oberhalb der Geraden liegt. Dies könnte schon das gesuchte Kriterium darstellen, erweist sich jedoch als rechnerisch zu aufwendig. Darum formen wir die drei Beziehungen elementar um zu:

$$(1a)\quad n_x(P)\,\Delta Y - n_y(P)\,\Delta x = 0$$

$$(2a)\quad n_x(P)\,\Delta Y - n_y(P)\,\Delta x > 0$$

$$(3a)\quad n_x(P)\,\Delta Y - n_y(P)\,\Delta x < 0$$

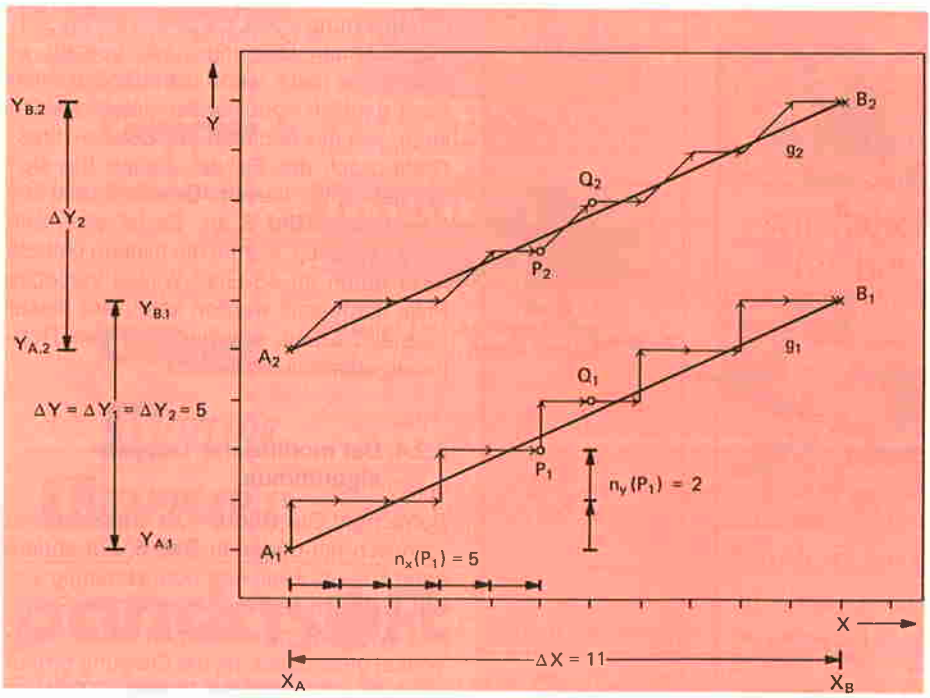


Bild 4

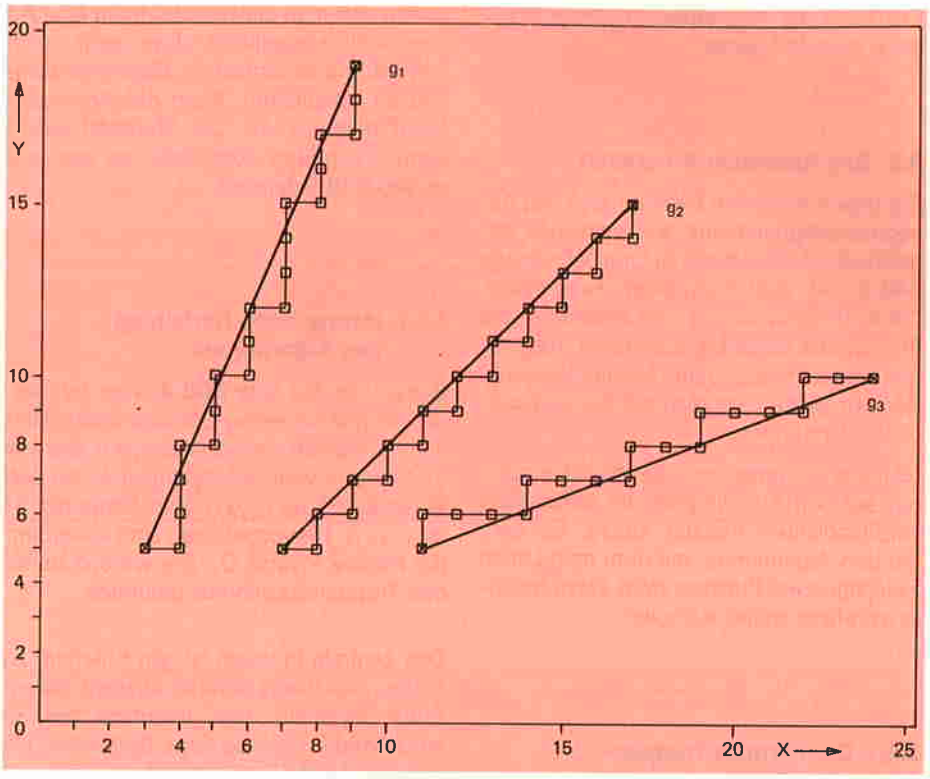


Bild 5

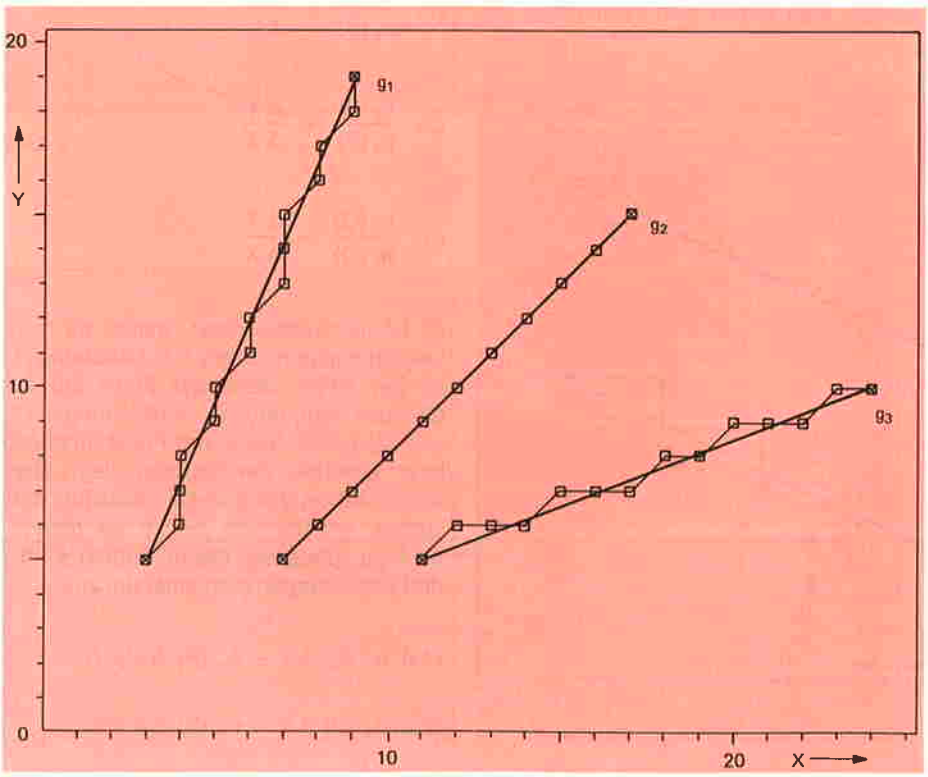


Bild 6

Das sieht schon viel besser aus: Nennen wir den Term auf der linken Seite einmal D. Wenn man von diesem D bei jedem Zeichenschritt in y-Richtung die Differenz der x-Komponenten von Anfangs- und Endpunkt subtrahiert, bei jedem Schritt in x-Richtung jedoch die der y-Komponenten dazuaddiert, so ist D dann und nur dann gleich null, wenn der gerade gezeichnete Punkt auf der Geraden liegt. D ist kleiner als null, wenn der Punkt unterhalb, größer als null, wenn der Punkt oberhalb der Geraden liegt. Ich formuliere den Algorithmus:

Führe beim Zeichnen zusätzlich eine Saldovariable D. Setze D beim Beginn jede Gerade auf 0. Subtrahiere von D bei jedem Schritt in y-Richtung den Betrag von Δ_x, addiere zu D bei jedem Schritt in x-Richtung den Betrag von Δ_y. Ist nun D größer als null, gehe beim nächsten Schritt in y-Richtung, ist D kleiner als null, gehe in x-Richtung. Wiederhole dies bis zum Endpunkt. Das ist im wesentlichen der Treppenalgorithmus, mit dem bespielsweise Applesoft arbeitet. Hier betrachten wir nur Geraden einer speziellen Orientierung. Die o.g. Beziehungen sind jedoch auch auf allgemeine Geraden anwendbar. Man betrachtet die Geraden, als seien sie wie die Geraden in **Bild 4** orientiert, und führt unabhängig davon die echten Bildschirmkoordinaten.

3.2.3. Nachteile des Treppenalgorithmus

Der oben vorgestellte Algorithmus mag angehen, wenn man beispielsweise über die Auflösung eines Hardwareplotters verfügt. Meiner Meinung nach versagt er spätestens dann, wenn man die einzelnen Pixel deutlich voneinander unterscheiden kann, wie das auch bei der Double-Hires-Grafik noch der Fall ist. Sehen Sie sich speziell unter diesem Gesichtspunkt die Geraden in **Bild 5** an. Es ist eigentlich nicht einsichtig, warum die mittlere Gerade nicht durch im 45-Grad-Winkel versetzte Pixel dargestellt werden kann. Auf dieser Idee aufbauend, erweiterte ich den Treppenalgorithmus ein wenig.

3.2.4. Der modifizierte Treppenalgorithmus

Betrachten Sie **Bild 6**! Die Geraden sind identisch mit denen in **Bild 5**, nur anders approximiert. Hier wird nun, abhängig von der Geradensteigung, statt einem Schritt in x-(y-)Richtung ein Schritt im 45-Grad-Winkel gezeichnet. Ist die Steigung größer als 45 Grad, wird der Schritt in x-Richtung ersetzt, sonst der Schritt in y-Richtung.

Checkmate Technology

™ **Speichererweiterungen**
mit Zukunft
für Apple IIe und Apple IIc

Multiram Leistungsmerkmale:
- 16bit CPU Port
- 65C816 Coprozessor, der den Multiram Speicher linear addressieren kann (sofort lieferbar)
- inclusive Apple Works Memory Expander (endlich genug Speicherplatz für Appleworks)
- inkl. Ramdisk Software für DOS und ProDos
- Ramdisk für Pascal und CP/M optional

speziell beim Apple IIe:
- auf einer Karte bis max. 1MB erweiterbar mit Huckepackkarte auf 1,7 MB erweiterbar
- kommt in Auxiliary Slot (3)
- ersetzt erweiterte 80 Zeichen Karte
- RGB Farboption lieferbar

speziell beim Apple IIc:
- bis 512k erweiterbar
- ohne externes Laufwerk echt portabel

pandasoft Dr.-Ing. Eden

Kataloganforderung und Bestellung: Tel.: 030/31 04 23 · Telex 185 859
Uhlandstraße 195 · D-1000 Berlin 12

kyan PASCAL

Pascal Compiler für Apple II (+,e,c)
- erzeugt 6502 Assembler-Code
- benötigt keine Z-80 Karte
- läuft unter Prodos
- integrierter Assembler
- inclusive Editor (full screen)
- mehrfach in Peeker getestet
- neu: Kix eine Uni™ ähnliche Betriebssystem Umgebung

Kennenlernaktion bis 30. Juni:
Kyan 2.0 inkl. KIX: DM 198,–

nach Ablauf der Kennenlernaktion:
Kyan 2.0 alleine DM 198,00
Kyan 2.0 inkl. Kix: DM 278,00

Unix ist ein eingetragenes Warenzeichen der Bell Laboratories

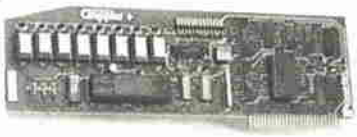
pandasoft Dr.-Ing. Eden

Kataloganforderung und Bestellung: Tel.: 030/31 04 23 · Telex 185 859
Uhlandstraße 195 · D-1000 Berlin 12

Orange™ Printer Interface **Druckerinterfaces** für Apple II+/e/c/III Interfaces auf dem **neuesten Stand der Technik. Kompatibel** mit allen gängigen Druckern wie: APPLE, EPSON, STAR, NEC, OKIDATA usw. Passende Treibersoftware wird über Dip-Switch ausgewählt.

Grappler + Printer Interface **Grafikfähiges Druckerinterface** das keine Wünsche mehr offen läßt. Über **2 Dutzend Kommandos** ermöglichen die volle Kontrolle über alle Möglichkeiten Ihres Druckers. Jetzt auch mit **IIe Features: Double Hires Grafics** und **80 Zeichen Dump** mittels Druckerpuffer nachrüstbar über Bufferboard.

BUFFERED Grappler + Besitzt alle Vorzüge des Grappler +, hat aber zusätzlich einen integrierten **16 K Druckpuffer,** der auf **32 oder 64 K aufrüstbar** ist.



SERIAL Grappler Printer Interface **Serielles Druckerinterface** speziell für den **Apple Imagewriter.**

HOTLINK Seriell-nach-Parallel-Wandler für den IIc im Kabel integriert.

GRAPPLER C wie Hotlink, jedoch zusätzlich Imagewriter Emulation und Grafik Software-Diskette.

pandasoft Dr.-Ing. Eden

Kataloganforderung und Bestellung: Tel.: 030/31 04 23 · Telex 185 859
Uhlandstraße 195 · D-1000 Berlin 12

Sie haben einen Apple ...

wir haben die Software ... | und die Hardware ...

wir haben die Bücher ... | und die Zeitschriften *...

*Fordern Sie unseren Gratiskatalog an!

ALLES FÜR DEN APPLE II+, IIe, IIc UND MACINTOSH

pandasoft Dr.-Ing. Eden
Uhlandstraße 195 · D-1000 Berlin 12
Tel.: 030/31 04 23 · Telex 185 859

Ich besitze einen Apple II+, IIe, IIc. Bitte schicken Sie mir Ihren 168 Seiten Apple II Katalog
Name Adresse

Das läßt sich durch einen einmaligen Vergleich von Δ_x mit Δ_y erledigen. Ist Δ_y größer als Δ_x, so ist die Geradensteigung größer als 45 Grad.

3.2.5. Implementierung in Assembler

Die Zeilen 517 bis 753 des Assemblerlistings sind abgedruckt. Der Einstieg in die Routine zum Geradenzeichen liegt in LINE. Zuerst werden die Differenzen der Anfangs- und Endkoordinaten berechnet, anschließend wird, falls nötig, durch Zweierkomplementierung der Absolutbetrag der Differenz gebildet. Ist die Differenz negativ, so wird gleichzeitig durch negatives XADD bzw. YADD die negative Laufrichtung festgelegt.
Sind die Differenzen bestimmt und die Laufrichtungen festgelegt, wird überprüft, ob eine Koordinatendifferenz identisch null ist. In diesem Fall wird die Gerade in XLINE bzw. YLINE getrennt weiterbearbeitet.
Nun werden die Differenzen verglichen. Ist DX größer als DY, wird in der Variablen FLAG Bit 7 gesetzt. Die Saldovariable wird auf 0 gesetzt, ein Schrittzähler wird initialisiert. Dann erfolgt in NOTREADY der Einstieg in die Zeichenschleife. Anschließend wird entschieden, ob der nächste Schritt in x- oder y-Richtung gehen soll. In XSTEP und YSTEP wird nun wie beschrieben die Saldovariable D aktualisiert, dann werden die nötigen Manipulationen am Grafikspeicher vorgenommen, auf die ich hier nicht weiter eingehen möchte; sie sind in Peeker, 2/84 beispielsweise hinreichend beschrieben. Sind die Manipulationen vorgenommen, wird nachgeprüft, ob ein Doppelschritt fällig ist. In diesem Fall wird nicht gezeichnet, sondern erst der zweite Schritt abgewartet. Ist schließlich der Schrittzähler bei null angelangt, ist die Gerade fertig approximiert.

4. PLOT.3.E und SUPERDUMP

Läuft SUPERDUMP, das für Epson-FX80 bzw. Imagewriter I vorgesehen ist, auf Ihrem Drucker? Wenn ja, haben Sie ausgesorgt. Ändern Sie die im Listing unter SUPERDUMP.PATCH angegebenen Programmzeilen im SUPERDUMP.EPSON bzw. SUPERDUMP.IMAGEWRITER. Besitzen Sie die Sammeldiskette #5 (mit Aufsatz in Heft 6/85), haben Sie es noch einfacher. Laden Sie das Applesoft-Programm und geben Sie EXEC SUPERDUMP.PATCH ein. Speichern Sie das gepatchte Programm unter neuem Namen ab, fertig. Zur Bedienung von SUPERDUMP lesen Sie bitte Peeker *6/85*.

PLOT.3.E ist in der Standardversion darauf eingerichtet, mit dem modifizierten SUPERDUMP zusammenzuarbeiten. Die PLOT.3.E Grafikfiles können vom SUPERDUMP gelesen werden, PLOT.3.E kann direkt SUPERDUMP aufrufen, das modifizierte SUPERDUMP ruft wieder PLOT.3.E auf. Benutzen Sie eine RAM-Disk, gibt es beim Laden der Programme fast keine Zeitverzögerungen.
Anm.d.Red.: Es sei hier noch einmal darauf hingewiesen, daß das Applesoft-Programm SUPERDUMP einschließlich der Variablen genau vor der Hires-Seite 1 endet. Wenn Sie also SUPERDUMP auch nur geringfügig ändern oder erweitern wollen, so entfernen Sie zunächst mit dem Applesoft-MACROEDITOR (Sammeldisk #16) über das &K-Kommando alle REMs, womit SUPERDUMP erheblich kürzer wird und damit keine Speicherkonflikte auftreten können.

5. Anpassung von PLOT.3.E an andere Gerätekonfigurationen

Unter PLOT.PATCH finden Sie die Modifikationen, die Sie machen müssen, um auf einem EPSON FX-80/85 mit Grafikinterface direkt die Grafiken auszudrucken. Sie sparen sich dann den Umweg über SUPERDUMP und erhalten außerdem ein ausführliches Protokoll mit Funktion, Darstellungsgrenzen und Parameterwerten.
Inzwischen scheint sich ProDOS als neues Standardbetriebssystem durchzusetzen. Unverträglichkeiten mit PLOT.3.E bestehen zwar keine, doch sind Änderungen speziell im Disk-Menü angebracht. Es müßte derart modifiziert werden, daß statt der Slot-Drive-Kombination ein PREFIX einzugeben ist. Sollte Interesse daran bestehen, bin ich gerne bereit, einen PATCH-File in einer der nächsten Peeker-Ausgaben zu veröffentlichen, der dies bewerkstelligt.

6. Kurzhinweise

1. Zweck:
Erstellung grafischer Darstellungen von Funktionen einer reellen Veränderlichen in doppelt hochauflösender Grafik
2. Konfiguration:
Apple IIe mit 64K-Karte oder IIc; DOS 3.3; Drucker optional, SUPERDUMP empfehlenswert
3. Test:
EXEC PLOT.3.PRO
4. Sammeldisk:
PLOT.3.PRO
PLOT.3.E
PLOT.BX
T.PLOT.BX
PLOT.HELP.1
PLOT.HELP.2
PLOT.HELP.3
PLOT.HELP.4
PLOT.HELP.5
PLOT.PATCH
SUPERDUMP.PATCH

Hinweise für „Eintipper"

PLOT.3.E läßt sich wie üblich eingeben. Schwieriger ist es dagegen, die einzelnen „PLOT.HELP."-Files einzugeben. Da Applesoft Kommata im INPUT-Befehl nicht zuläßt, muß *jede Zeile* mit einem Gänsefüßchen (" = CHR$(34)) beginnen! Die Zeilen sollten nicht länger als 79 Zeichen sein! Besitzen Sie keinen Texteditor, müssen Sie die einzelnen Zeilen als DATA-Statements in ein BASIC-Programm eingeben, das dann später diese als Strings mit vorangestelltem " in den Textfile überträgt. Sind Sie damit fertig, dürfen Sie „zur Erholung" den Hexdump der Assemblerroutinen eintippen.

7. Literaturhinweise

–PLOT.2.0, Peeker 5/85, S. 17ff
–SUPERDUMP von D. Geiß, Peeker 6/85, S. 22ff
–Apple DOS 3.3 von U. Stiehl, S. 41-43
–Mikrocomputer-Grafik von R. E. Myers

8. PLOT-Hilfemenüs

Wichtig: Vergessen Sie nicht, am Ende jedes Helpfiles die Marke **.EOF** anzuhängen. Sonst findet PLOT.3.E nicht das Ende des Files, und das Programm kann mit einer Fehlermeldung abgebrochen werden!

PLOT.HELP.1

PLOT.3.E – Ein Funktionsplotprogramm

Dieses Programm nennt sich PLOT.3.E. Es stellt Funktionen einer reellen Veränderlichen auf dem Bildschirm der Apple-Computer-Typen //e und //c grafisch dar.
Da dabei die sogenannte doppelt hochauflösende Grafik dieser Rechner verwendet wird, müssen Sie zusätzlich im Apple //e eine erweiterte 80-Zeichenkarte installieren. Der Apple //c benötigt keine zusätzliche Hardware. Um eine befriedigende Auflösung zu erreichen, ist jedoch ein Monochrom-Monitor empfehlenswert.
Ist Ihr Computer zusätzlich an einen grafikfähigen Drucker angeschlossen, können Sie mit dem Programm SUPERDUMP (Peeker 6/85, Sammeldiskette #5) Ihre Grafiken ausdrucken. Beachten Sie jedoch bitte zusätzlich den zu PLOT.3.E gehörigen Artikel, Abschnitt 'PLOT .3.E und SUPERDUMP'.
Die Funktion wird in einem Achsenkreuz gezeichnet, die Achsen werden mit Skalenmarkie-

rungen versehen. Auf dem längeren Ast jeder Achse werden mindestens zwei, maximal 19 Markierungen angebracht. Den Abbildungsmaßstab können Sie relativ frei festlegen. Dies geschieht, indem Sie die von Ihnen gewünschten Darstellungsgrenzen eingeben. Bei deren Eingabe müssen Sie lediglich darauf achten, daß die 0 im Darstellungsintervall jeder Achse vorkommt. Sonst könnte kein Achsenkreuz gezeichnet werden.
Natürlich müssen Sie auch die zu zeichnende Funktion eingeben. Diese muß als korrekte Applesoft-BASIC-Funktion eingegeben werden. Sollten Sie sich darin nicht auskennen, rufen Sie bitte im Programmteil 'Funktionseingabe' die dazu bereitgehaltenen Informationen ab.
Punkte, an denen die Funktion nicht definiert ist (z.B. Polstellen), werden intern abgefangen, das Zeichnen wird solange unterbrochen, bis die Funktion wieder reelle Werte annimmt (die im Darstellungsintervall liegen).
Zusätzlich zur unabhängigen Variablen X können Sie in der Funktion die Parameter R,S und T verwenden. Sie können die Funktion mehrmals mit verschiedenen Parameterwerten zeichnen lassen. So können Sie ganze Funktionenscharen darstellen lassen. T nimmt unter den Parametern eine Sonderstellung ein. Wenn T mit einer festen Schrittweite zu- oder abnehmen soll, können Sie sich die ständige Neueingabe sparen und statt dessen nur Anfangs-, Endwert und Schrittweite festlegen. Die Funktionsschar wird dann automatisch gezeichnet.
Wenn Sie die Eulersche Zahl e oder die Kreiszahl pi in der Funktion benötigen, brauchen Sie diese nicht im ganzen einzutippen. Im Programm sind die Variablen PI und E vordefiniert. Den dekadischen Logarithmus können Sie als FN LG(X) aufrufen.
Während des Zeichenvorgangs können Sie durch Betätigen der Leertaste auf Textdarstellung umschalten. Dort finden Sie alle Ihre Eingaben nochmals aufgeführt. Nochmaliges Betätigen der Leertaste läßt die Grafik wieder erscheinen. Stellen Sie beim Zeichnen fest, daß die Grafik nicht so wird, wie Sie es sich vorstellten, können Sie durch gleichzeitiges Betätigen der 'Ctrl'- und der 'C'-Taste den Zeichenvorgang abbrechen. Das Programm springt dann ins Hauptmenü, das sonst erst erreicht wird, wenn die Grafik fertig gezeichnet ist.
Von diesem Programmteil aus können Sie alle Funktionen von PLOT.3.E aufrufen (Neueingabe von Funktion, Parametern und Darstellungsgrenzen, Löschen des Bildschirms, Darstellung aller eingegebenen Daten, von neuem zeichnen und auf die Diskettenlaufwerke zugreifen).
.EOF

PLOT.HELP.2

PLOT.3.E – Eingabe der Darstellungsgrenzen

Bei der Eingabe der Darstellungsgrenzen haben Sie die Wahl:
Entweder Sie geben die Darstellungsintervalle beider Funktionen explizit ein. Das ist sicher der einfachste Weg, hat jedoch auch seine Nachteile. Oder Sie geben die Darstellungsgrenzen für eine Achse ein, für die andere Anfangs- und Endwert 0. Dann wird in x- und y-Richtung mit dem gleichen Maßstab abgebildet.
Gegen die erste Möglichkeit spricht, daß die Abbildungsmaßstäbe der beiden Achsen absolut nichts miteinander zu tun haben. Nur wenn X- und Y-Achse auf denselben Maßstab abbilden, erkennt man solche Feinheiten, wie etwa, daß die Sinusfunktion in den Nullstellen die Steigung +1 oder -1 hat. Gegen die zweite Möglichkeit spricht jedoch, daß dann die beiden nicht eingegebenen Grenzen symmetrisch zum Ursprung festgelegt werden. Das kann gut gehen, das Ergebnis ist jedoch oft nicht allzu befriedigend.
An dieser Stelle noch einmal die Bitte: Sorgen Sie dafür, daß in jedem Darstellungsintervall die 0 liegt oder zumindest als Grenze eingegeben wird. Da sonst kein Achsenkreuz gezeichnet werden könnte, setzt das Programm für diesen Fall die betragsmäßig kleinere Grenze auf 0!
Die Eingabeprozedur selbst läuft wie folgt ab:
Zuerst werden Sie gebeten, den Anfangswert für die X-Achse anzugeben. Tun Sie dies, beachten Sie dabei jedoch die amerikanische Notation ('.' statt Dezimalkomma). Dann schließen Sie bitte die Eingabe mit der 'Return'-Taste ab. Genauso geht es mit dem Endwert der X-Achse und dann mit Anfangs- und Endwert der Y-Achse weiter. Anschließend werden Ihre Eingaben analysiert. Dabei werden eventuell vertauschte Anfangs- und Endwerte ausgetauscht, dann wird geprüft, ob der Anfangswert jeder Achse kleiner oder gleich 0 und ob der Endwert jeder Achse größer oder gleich 0 ist. Ist dies nicht der Fall, wird der falsche Wert auf 0 gesetzt. Anschließend wird überprüft, ob für eine Achse Anfangs- und Endwert 0 eingegeben wurden. Ist dies der Fall, so werden die Darstellungsgrenzen derart festgelegt, daß X- und Y-Achse den gleichen Maßstab besitzen.
Haben Sie die Darstellungsgrenzen schon einmal eingegeben, sind jedoch mit dem Ergebnis nicht ganz zufrieden, so haben Sie wiederum die Wahl:
Sie können entweder die Darstellungsgrenzen komplett neu eingeben oder die Achsen direkt manipulieren. Näheres zur letzten Möglichkeit entnehmen Sie bitte den getrennt dazu abrufbaren Informationen.
.EOF

PLOT.HELP.3

PLOT.3.E – Achsenmanipulation

Sie haben folgende Möglichkeiten der direkten Achsenmanipulation:
1 - Achsen strecken oder stauchen
2 - Manipulation des Verhältnisses der positiven zur negativen Halbachse
Zu 1: Nehmen wir einmal an, die Grafik sei schon ganz ansprechend, nur Sie wünschen eine höhere Aufspreizung der Grafik in y-Richtung. Sagen wir, Sie wollen um einen Faktor 2 spreizen. Dann haben Sie entweder die Möglichkeit, die bisherigen Darstellungsgrenzen für die y-Achse im Kopf durch 2 zu dividieren und neu einzutippen. Das muß aber nicht sein. Ebenso können Sie, nachdem Sie diese Erläuterungen gelesen haben, auf die Frage 'Y-Achse strecken?' mit 'J' antworten. Dann müssen Sie nur den Streckfaktor angeben, und zwar nicht als unhandlichen Dezimalbruch, sondern fast als echten Bruch. Wenn Sie nun auf die Frage 'Verhältnis alte ...' 'zu neuen Grenzen' 2 und 1 eingeben, werden Sie anschließend angenehm überrascht feststellen, daß das Programm Ihnen das Kopfrechnen abgenommen hat.
Das ist der ganze Trick. Wie wertvoll diese Option ist, werden Sie feststellen, wenn Sie einmal versuchen, zwei siebenstellige Zahlen im Verhältnis 17 zu 13 zu teilen. Viel Vergnügen...
Zu 2: Nun stellen Sie sich vor, Sie lassen eine Funktion in einem symmetrischen Achsenkreuz zeichnen. Sie stellen aber beim Zeichnen fest, daß das untere Viertel des Bildschirms nicht genutzt wird. Nützlich wäre es da, wenn man einfach dem Computer sagen könnte: 'Hör 'mal, der Maßstab war ja ganz gut, schiebe doch beim nächsten mal die X-Achse einfach um ein Viertel oder so nach unten.' Genau das können Sie bei PLOT.3.E auch. Dies geschieht, indem Sie angeben, in welchem Verhältnis der positive und der negative Teilast einer Achse zueinander stehen sollen. Bei unserem Beispiel wäre das ein Verhältnis von 2:1. Eingegeben wird das ganze wie oben bei der Streckung der Achsen.
Das hörte sich jetzt alles komplizierter an, als es in Wirklichkeit ist. Probieren Sie es am besten mal aus. Wenn es nicht klappt, gehen Sie die Beispiele Schritt für Schritt durch, die im zu diesem Programm gehörenden Artikel aufgeführt sind. Wie gesagt, so kompliziert ist's nicht.
.EOF

PLOT.HELP.4

PLOT.3.E – Funktionseingabe

Computer sind in mancher Hinsicht ziemliche Dummköpfe, gepaart mit dem Hang zu einer gewissen Dickschädeligkeit. Darum geben Sie es auf, dem Computer eine Funktion 'f(x) = sin x' vorzusetzen. Ein wahrer Apple moniert da stur einen '? SYNTAX ERROR'. (Den nun folgenden Abschnitt können Besitzer eines Apple //c oder Apple //e mit neuen ROMs überspringen)
Zuallererst drücken Sie die 'Caps Lock'-Taste (unten links auf der Tastatur) so, daß sie einrastet. Applesoft-BASIC akzeptiert in arithmetischen Ausdrücken einzig und allein Großbuchstaben.
In BASIC gibt es folgende Verknüpfungszeichen:
+ : Addition oder positives Vorzeichen
- : Subtraktion oder negatives Vorzeichen
* : Multiplikation
/ : Division
Applesoft-BASIC verlangt, daß alle Multiplikationen ausgeschrieben werden. Ein Ausdruck wie '3(X-T)' ist für Sie das Einfachste auf der Welt, für den Computer schlicht und ergreifend ein Syntax-Fehler. Ebenso gilt ein Divisionszeichen nur für den nächsten Wert, im Gegensatz zum Bruchstrich. Wollen Sie längere Brüche ausdrücken, müssen Sie Zähler und Nenner einklammern. Im Zweifelsfalle verwenden Sie lieber ein Klammernpaar mehr.
Es gibt folgende eingebauten Funktionen:
SIN (X) : Sinus von X, Argument im Bogenmaß
COS (X) : Cosinus von X, Argument im Bogenmaß
TAN (X) : Tangens von X, Argument im Bogenmaß

ATN (X) : Arcustangens von X, Ergebnis im Bogenmaß
SQR (X) : Quadratwurzel aus X
X ↑ R : von X wird die R-te Potenz gebildet
X↑ (1/R) : aus X wird die R-te Wurzel gezogen
EXP (X) : Exponentialfunktion zur Basis e
LOG (X) : natürlicher Logarithmus (Basis e)
INT (X) : (Gauß-Klammer) liefert größte ganze Zahl, die kleiner als X ist
ABS (X) : Absolutbetrag von X
SGN (X) : Signumfunktion. Liefert 0 für X=0, +1 für X>0, -1 für X<0
So nun sollen Sie noch sehen, wie eine 'legale' Funktion aussieht: Das Programm gibt Ihnen 'Y= ' vor. Nun müssen Sie nur noch nach den obenstehenden Regeln eine Funktion zusammenbasteln. Das sieht dann so aus:
Y= SIN(X) * COS(X)
Y= -(X - 1)/(X↑2 - X + 3)
Y= T/X (übrigens, T ist so ein mysteriöser Scharparameter)
Y= INT(X) (Treppenfunktion)
Y= X - INT(X) (Sägezahnfunktion)
Y= LOG(X)/LOG(R) (liefert Logarithmus zur Basis R)
Andere Beispiele finden Sie im zu diesem Programm gehörenden Artikel.

.EOF

PLOT.HELP.5

PLOT.3.E – Diskmenü

Sie können vom Disk-Menü aus folgende Diskettenoperationen ausführen:
1. Das Inhaltsverzeichnis der Diskette abrufen.
2. Grafikfiles auf Diskette abspeichern und von Diskette laden.
3. Erläuterungen zum Programm abrufen. Das tun Sie gerade.
4. Das aktive Laufwerk auswählen.
5. Je nach Version können Sie Superdump aufrufen oder direkt die Grafik auf einem grafikfähigen Matrixdrucker ausdrucken lassen.
ad 2. Eine Grafik wird auf Diskette in zwei getrennten Binärfiles abgespeichert. Dabei werden zu den von Ihnen eingebenen Filenamen die Suffixe '.M' und '.A' hinzugefügt. Wenn Sie diese Files wieder einlesen wollen, so lassen Sie bitte die Suffixe weg.
Diese Option wurde vor allem deshalb eingerichtet, damit Sie Grafiken, die mit PLOT.3.E erstellt wurden, mit dem Programm Superdump (in Peeker 6/85 und auf Sammeldisk #5) auf einem Matrixdrucker ausdrucken können.
ad 4. Mit dieser Option können Sie jede Slot/Drivekombination auswählen, die sinnvoll ist. Logischerweise ist Slot 0 ausgeschlossen. Sinn dieser Option ist es, Grafikfiles auf einer schnellen RAM-Disk abzuspeichern, dann Superdump zu laden und die Grafik wieder in den Speicher zurückzuholen. So kommen Sie ohne nennenswerten Zeitverlust zu Ihren Grafik-Ausdrucken. Sind Sie im Besitz eines Druckerinterfaces mit Double-Hires-Software, so ändern Sie bitte PLOT.3.E wie in Peeker 6/86 beschrieben ab. Dann erstellt PLOT.3.E direkt einen Ausdruck der Grafik, zusätzlich jedoch werden die Funktion, der Darstellungsbereich und alle verwendeten Parameter ausgedruckt.

.EOF

PLOT.BX (Hexdump)

```
BSAVE PLOT.BX, A$ØBØØ, L$Ø54D

$ØBØØ: 48 A9 65 8D F6 Ø3 A9 ØC
$ØBØ8: 8D F7 Ø3 68 6Ø A5 B8 48
$ØB1Ø: A5 B9 48 A9 59 2Ø 5C DB
$ØB18: A2 BD 2Ø 2E D5 2Ø 6C D5
$ØB2Ø: AØ ØØ 84 B8 A9 Ø9 85 B9
$ØB28: B9 Ø7 ØC FØ Ø5 91 B8 C8
$ØB3Ø: DØ F6 B9 F6 Ø1 91 B8 FØ
$ØB38: Ø3 C8 DØ F6 AØ ØØ 8C B9
$ØB4Ø: ØB 84 FF A9 ØA 8D BA ØB
$ØB48: B9 FD Ø1 FØ 38 3Ø Ø9 2Ø
$ØB5Ø: BF ØB 2Ø B8 ØB C8 DØ FØ
$ØB58: 48 A9 2Ø 2Ø B8 ØB 38 68
$ØB6Ø: E9 C2 9Ø 15 AA BD DD ØB
$ØB68: AA BD E9 D1 1Ø Ø1 18 29
$ØB7Ø: 7F 2Ø B8 ØB E8 BØ F2 9Ø
$ØB78: DC A2 FD A9 3F 2Ø B8 ØB
$ØB8Ø: E8 DØ FA FØ DØ 2Ø B1 ØØ
$ØB88: 2Ø 13 E3 68 85 B9 68 85
$ØB9Ø: B8 2Ø B1 ØØ 2Ø BE DE 2Ø
$ØB98: E3 DF 24 11 1Ø 15 AØ ØØ
$ØBAØ: AD B9 ØB 91 83 C8 A9 ØØ
$ØBA8: 91 83 C8 A9 ØA 91 83 2Ø
$ØBBØ: A9 ØC 6Ø A2 A3 4C 12 D4
$ØBB8: 8D ØØ ØA EE B9 ØB 6Ø Ø8
$ØBCØ: A2 Ø2 DD D7 ØB FØ Ø5 CA
$ØBC8: 1Ø F8 28 6Ø 48 BD DA ØB
$ØBDØ: Ø5 FF 85 FF 68 28 6Ø 52
$ØBD8: 53 54 Ø1 Ø2 Ø4 ØØ Ø2 Ø6
$ØBEØ: ØA ØC ØF 13 14 15 16 17
$ØBE8: 18 1B 1D 1E 1F 2Ø 23 26
$ØBFØ: 29 2C 2F 34 37 3A 3D 4Ø
$ØBF8: 43 46 49 4C 4F 52 56 59
$ØCØØ: 5D 6Ø 63 67 6C 72 76 B8
$ØCØ8: C2 59 28 58 29 DØ ØØ 68
$ØC1Ø: A8 68 A6 DF 9A 48 98 48
$ØC18: 6Ø 2Ø B1 ØØ 2Ø BB DE 2Ø
$ØC2Ø: F8 E6 EØ 5Ø BØ 5D 8E 7B
$ØC28: Ø5 2Ø B7 ØØ 2Ø BE DE 2Ø
$ØC3Ø: F8 E6 EØ 18 BØ 4D 8A 2Ø
$ØC38: 5B FB 2Ø A9 ØC 2Ø 95 D9
$ØC4Ø: 6Ø 4C ØD ØB 2Ø 3F ØE 2Ø
$ØC48: A9 ØC 2Ø 95 D9 6Ø 2Ø 3F
$ØC5Ø: ØE 2Ø 1C ØE 2Ø A9 ØC 2Ø
$ØC58: 95 D9 6Ø 2Ø 52 ØE 2Ø A9
$ØC6Ø: ØC 2Ø 95 D9 6Ø 2Ø 9F ØC
$ØC68: C9 93 FØ 47 C9 91 FØ DE
$ØC7Ø: C9 9Ø FØ DØ C9 89 FØ E3
$ØC78: C9 D9 FØ 9D C9 C2 FØ C1
$ØC8Ø: 4C C9 DE 4C 99 E1 2Ø 46
$ØC88: E7 EØ CØ BØ F6 A4 51 A5
$ØC9Ø: 5Ø CØ Ø2 9Ø Ø6 DØ EC C9
$ØC98: 3Ø BØ E8 6Ø ØØ ØØ ØØ 8D
$ØCAØ: 9C ØC 8E 9D ØC 8C 9E ØC
$ØCA8: 6Ø AD 9C ØC AC 9E ØC AE
$ØCBØ: 9D ØC 6Ø 2Ø B1 ØØ C9 C1
$ØCB8: FØ ØD 2Ø 86 ØC 2Ø 62 ØE
$ØCCØ: 2Ø B7 ØØ C9 C1 DØ E2 2Ø
$ØCC8: B1 ØØ 2Ø 86 ØC 8E DE ØC
$ØCDØ: 8D DC ØC 8C DD ØC 2Ø ØD
$ØCD8: ØD 4C CØ ØC ØØ ØØ ØØ 6Ø
$ØCEØ: A9 ØØ E5 D4 AA A9 ØØ E5
$ØCE8: D5 85 1E FØ F2 A9 FF 85
$ØCFØ: D1 85 FF 4C 81 ØD A9 ØØ
$ØCF8: E5 D7 AA A9 FF 85 1E A9
$ØDØØ: Ø1 85 D1 85 FF DØ EC 38
$ØDØ8: BØ D6 38 BØ E9 38 AD DE
$ØD1Ø: ØC E5 E2 85 D7 BØ ØD 38
$ØD18: A9 ØØ E5 D7 85 D7 A9 FF
$ØD2Ø: 85 E6 3Ø Ø4 A9 Ø1 85 E6
$ØD28: 38 AD DC ØC E5 EØ 85 D4
$ØD3Ø: AD DD ØC E5 E1 85 D5 BØ
$ØD38: 15 38 A9 ØØ E5 D4 85 D4
$ØD4Ø: A9 ØØ E5 D5 85 D5 A9 FF
$ØD48: 85 E4 85 E5 3Ø Ø8 A9 ØØ
$ØD5Ø: 85 E5 A9 Ø1 85 E4 A5 D7
$ØD58: FØ AD A5 D4 DØ Ø4 A5 D5
$ØD6Ø: FØ A8 38 A5 D5 DØ Ø4 A5
$ØD68: D4 C5 D7 6A 85 FF 38 A9
$ØD7Ø: ØØ 85 DØ 85 D1 E5 D7 38
$ØD78: E5 D4 AA A9 FF E5 D5 85
$ØD8Ø: 1E A5 D1 3Ø 5A DØ Ø8 A5
$ØD88: DØ DØ Ø4 24 FF 1Ø 5Ø 18
$ØD9Ø: A5 E2 65 E6 85 E2 A8 B9
$ØD98: 96 ØE 85 CE B9 56 ØF 85
$ØDAØ: CF 38 A5 DØ E5 D4 85 DØ
$ØDA8: A5 D1 E5 D5 85 D1 E8 DØ
$ØDBØ: Ø4 E6 1E FØ Ø4 24 FF 3Ø
$ØDB8: 26 A5 E3 4A 9Ø Ø5 AC 54
$ØDCØ: CØ BØ Ø3 AC 55 CØ A8 B1
$ØDC8: CE Ø5 3Ø 91 CE 24 1E 3Ø
$ØDDØ: BØ 8D 54 CØ AD DD ØC 85
$ØDD8: E1 AD DC ØC 85 EØ 6Ø 18
$ØDEØ: A5 D2 65 E4 3Ø ØC C9 Ø7
$ØDE8: DØ 1Ø E6 E3 A9 ØØ 85 D2
$ØDFØ: FØ ØA C6 E3 A9 Ø6 85 D2
$ØDF8: DØ Ø2 85 D2 A4 D2 B9 16
$ØEØØ: 1Ø 85 3Ø E8 DØ Ø4 E6 1E
$ØEØ8: FØ AF A5 DØ 18 65 D7 85
$ØE1Ø: DØ 9Ø Ø2 E6 D1 24 FF 3Ø
$ØE18: AØ 4C 8F ØD A9 ØØ A8 85
$ØE2Ø: CE A9 2Ø 85 CF AD 55 CØ
$ØE28: A9 ØØ 91 CE AD 54 CØ A9
$ØE3Ø: ØØ 91 CE C8 DØ EF E6 CF
$ØE38: A5 CF C9 4Ø 9Ø E7 6Ø AD
$ØE4Ø: 52 CØ AD 57 CØ AD 5Ø CØ
$ØE48: 8D Ø1 CØ 8D ØD CØ AD 5E
$ØE5Ø: CØ 6Ø AD 56 CØ AD 51 CØ
$ØE58: AD 54 CØ 8D ØØ CØ 8D ØC
$ØE6Ø: CØ 6Ø 85 EØ 84 E1 86 E2
$ØE68: BD 96 ØE 85 CE BD 56 ØF
$ØE7Ø: 85 CF A5 EØ 2Ø 1D 1Ø A6
$ØE78: D2 BD 16 1Ø 85 3Ø A5 E3
$ØE8Ø: 4A 9Ø Ø5 AC 54 CØ BØ Ø3
$ØE88: AC 55 CØ A8 B1 CE Ø5 3Ø
$ØE9Ø: 91 CE 8D 54 CØ 6Ø ØØ ØØ
$ØE98: ØØ ØØ ØØ ØØ ØØ ØØ 8Ø 8Ø
$ØEAØ: 8Ø 8Ø 8Ø 8Ø 8Ø 8Ø ØØ ØØ
$ØEA8: ØØ ØØ ØØ ØØ ØØ ØØ 8Ø 8Ø
$ØEBØ: 8Ø 8Ø 8Ø 8Ø 8Ø 8Ø ØØ ØØ
$ØEB8: ØØ ØØ ØØ ØØ ØØ ØØ 8Ø 8Ø
$ØECØ: 8Ø 8Ø 8Ø 8Ø 8Ø 8Ø ØØ ØØ
$ØEC8: ØØ ØØ ØØ ØØ ØØ ØØ 8Ø 8Ø
$ØEDØ: 8Ø 8Ø 8Ø 8Ø 8Ø 8Ø 28 28
$ØED8: 28 28 28 28 28 28 A8 A8
$ØEEØ: A8 A8 A8 A8 A8 A8 28 28
$ØEE8: 28 28 28 28 28 28 A8 A8
$ØEFØ: A8 A8 A8 A8 A8 A8 28 28
$ØEF8: 28 28 28 28 28 28 A8 A8
$ØFØØ: A8 A8 A8 A8 A8 A8 28 28
$ØFØ8: 28 28 28 28 28 28 A8 A8
$ØF1Ø: A8 A8 A8 A8 A8 A8 5Ø 5Ø
$ØF18: 5Ø 5Ø 5Ø 5Ø 5Ø 5Ø DØ DØ
$ØF2Ø: DØ DØ DØ DØ DØ DØ 5Ø 5Ø
$ØF28: 5Ø 5Ø 5Ø 5Ø 5Ø 5Ø DØ DØ
$ØF3Ø: DØ DØ DØ DØ DØ DØ 5Ø 5Ø
$ØF38: 5Ø 5Ø 5Ø 5Ø 5Ø 5Ø DØ DØ
$ØF4Ø: DØ DØ DØ DØ DØ DØ 5Ø 5Ø
$ØF48: 5Ø 5Ø 5Ø 5Ø 5Ø 5Ø DØ DØ
$ØF5Ø: DØ DØ DØ DØ DØ DØ 2Ø 24
$ØF58: 28 2C 3Ø 34 38 3C 2Ø 24
$ØF6Ø: 28 2C 3Ø 34 38 3C 21 25
$ØF68: 29 2D 31 35 39 3D 21 25
$ØF7Ø: 29 2D 31 35 39 3D 22 26
$ØF78: 2A 2E 32 36 3A 3E 22 26
$ØF8Ø: 2A 2E 32 36 3A 3E 23 27
$ØF88: 2B 2F 33 37 3B 3F 23 27
$ØF9Ø: 2B 2F 33 37 3B 3F 2Ø 24
$ØF98: 28 2C 3Ø 34 38 3C 2Ø 24
$ØFAØ: 28 2C 3Ø 34 38 3C 21 25
$ØFA8: 29 2D 31 35 39 3D 21 25
$ØFBØ: 29 2D 31 35 39 3D 22 26
$ØFB8: 2A 2E 32 36 3A 3E 22 26
$ØFCØ: 2A 2E 32 36 3A 3E 23 27
$ØFC8: 2B 2F 33 37 3B 3F 23 27
$ØFDØ: 2B 2F 33 37 3B 3F 2Ø 24
$ØFD8: 28 2C 3Ø 34 38 3C 2Ø 24
$ØFEØ: 28 2C 3Ø 34 38 3C 21 25
$ØFE8: 29 2D 31 35 39 3D 21 25
$ØFFØ: 29 2D 31 35 39 3D 22 26
$ØFF8: 2A 2E 32 36 3A 3E 22 26
$1ØØØ: 2A 2E 32 36 3A 3E 23 27
$1Ø08: 2B 2F 33 37 3B 3F 23 27
$1Ø1Ø: 2B 2F 33 37 3B 3F Ø1 Ø2
$1Ø18: Ø4 Ø8 1Ø 2Ø 4Ø 85 D2 84
$1Ø2Ø: D3 A9 ØØ 85 E3 AØ Ø6 38
$1Ø28: A5 D2 F9 3F 1Ø AA A5 D3
$1Ø3Ø: F9 46 1Ø 9Ø Ø4 85 D3 86
$1Ø38: D2 26 E3 88 1Ø E9 6Ø Ø7
$1Ø4Ø: ØE 1C 38 7Ø EØ CØ ØØ ØØ
$1Ø48: ØØ ØØ ØØ ØØ Ø1
```

PLOT.3.E (Applesoft-Programm)

```
1000 GOTO 6000: REM Subroutinen folgen
1010 REM  Subroutine Zeichnen
1020 IF F$ = "" OR XL = YH AND YL = XH THEN  RETURN
1030 X0 = 0:ST = 1 / MX: REM  Initialisierung
1040 IF  RIGHT$ (P$,1) = "K" THEN  FOR T = T0 TO T1 STEP
     TS: REM  Starte T-Schleife
1050 GOSUB 3300:XA = XL: REM  Generiere Liste in Textseite
1060 X = XA: IF X > XH THEN 1160: REM Start x-Schleife
1070 IF  PEEK (KE%) > 127 THEN  GOSUB 1500: REM
     umschalten?
1080 IF  PEEK (PG%) THEN  &  POS (14,20): PRINT X;: CALL
     CL%: REM PRINT, wenn Textseite aktiv
1090 IF  NOT X0 THEN X0 =  FN XS(X):Y0 =  FN YS( FN Y(X)):
     GOTO 1160: REM Ausgangspunkt
1100 X1 =  FN XS(X):Y1 =  FN YS( FN Y(X)): REM Endpunkt
1110 IF Y1 < 0 THEN Y1 = 0: REM  gegen 'illegal Quantity'
1120 IF Y1 > 191 THEN Y1 = 191
1130 IF Y1 = 191 AND Y0 = 191 OR  NOT (Y1 OR Y0) THEN X0 =
     0: GOTO 1160: REM => Pol
1140 IF  NOT Y1 AND Y0 = 191 OR Y1 = 191 AND  NOT Y0 THEN
     X0 = 0: GOTO 1090: REM Keine Verbindung zwischen
     Polen
1150 &  HPLOT X0,Y0 TO X1,Y1:X0 = X1:Y0 = Y1: REM
     Zeichnen / neuer Ausgangspunkt
1160 IF X < XH THEN X = X + ST: GOTO 1070
1170 I = I + 1:P(0,I) = R:P(1,I) = S:P(2,I) = T: REM
     Parameterliste
1180 IF  RIGHT$ (P$,1) = "K" THEN X0 = 0: NEXT T: REM s.
     1030
1190 PRINT  CHR$ (7): GET A$: RETURN
1200 :
1290 REM  Fehlerbehandlung
1300 LN =  PEEK (218) + 256 *  PEEK (219):ER =  PEEK
     (222): CALL 3087
1310 IF ER = 133 OR ER = 69 OR ER = 53 THEN XA = ST + X:X0
     = 0: GOTO 1060
1320 REM Illegal Quantity, Overflow, Division by Zero
1330 IF ER = 255 THEN 6460: REM Hauptmenü
1340 IF ER = 254 THEN  PRINT : PRINT  CHR$ (7)"Bitte Zahl
     eingeben!": &  POS (CX%,CY%): RESUME : REM
     fehlerhafte Eingabe
1350 IF ER = 6 AND LN = 6060 THEN  TEXT : PRINT  CHR$
     (21): HOME : PRINT  CHR$ (7)"Achtung! File PLOT.BX
     fehlt!": PRINT "PLOT.3.E abgebrochen!" CHR$ (7): END
1360 IF ER > 0 AND ER < 16 THEN  TEXT : PRINT  CHR$ (21):
     HOME : PRINT  CHR$ (7)"DOS-Fehler #";ER;" in Zeile
     ";LN: END
1370 TEXT : PRINT  CHR$ (21): VTAB 20: PRINT  CHR$
     (7)"Fehler Code # ";ER;" in Zeile ";LN: END
1400 :
1490 REM Grafik/Text umschalten
1500 POKE  - 16368,0: REM  lösche Keyboardstrobe
1510 IF  NOT  PEEK (PG%) THEN  POKE  - 16303,0: POKE
     PG%,1: RETURN : REM Textseite
1520 POKE  - 16304,0: POKE PG%,0: RETURN : REM Grafikseite
1530 :
1690 REM  X-Achse strecken/stauchen
1700 CY% = 2:CX% = 1: &  POS (CX%,CY%): INPUT "Verhältnis
     alte Achse  ";X0
1710 CX% = 40: &  POS (CX%,CY%): INPUT "zu neue Achse
     ";X1
1720 RETURN
1730 :
1740 REM  Y-Achse strecken/stauchen
1750 CY% = 4:CX% = 1: &  POS (CX%,CY%): INPUT "Verhältnis
     alte Achse  ";Y0
1760 CX% = 40: &  POS (CX%,CY%): INPUT "zu neue Achse
     ";Y1
1770 RETURN
1780 :
1790 REM  Neue Aufteilung X-Achse
1800 CY% = 9:CX% = 1: &  POS (CX%,CY%): INPUT "Anteil für
     positiven ";MX
1810 CX% = 40: &  POS (CX%,CY%): INPUT "für negativen
     Achsenabschnitt ";BX
1820 RETURN
1830 :
1840 REM  Y-Achse neu teilen
1850 CY% = 11:CX% = 1: &  POS (CX%,CY%): INPUT "Anteil für
     positiven ";MY
1860 CX% = 40: &  POS (CX%,CY%): INPUT "für negativen
     Achsenabschnitt ";BY
1870 RETURN
1880 :
1890 REM Darstellungsgrenzen
1900 IF XL OR XH OR YL OR YH THEN  HOME : GOTO 2080: REM
     alte Werte zeigen
1910 HOME : PRINT "Bitte Darstellungsgrenzen eingeben!
     Wünschen Sie Erläuterungen? <J/N>";
1920 GOSUB 3400: IF A$ = "N" THEN  PRINT : VTAB 17: GOTO
     1950
1930 ID$ = "2": GOSUB 5510
1940 HOME : PRINT "Bitte Grenzen eingeben !"
1950 CY% = 6:CX% = 0: &  POS (CX%,CY%): CALL CL%: INPUT
     "Minimum für X: ";XL
1960 CX% = 40: &  POS (CX%,CY%): CALL CL%: INPUT "Maximum
     für X: ";XH
1970 CY% = 9:CX% = 0: &  POS (CX%,CY%): CALL CL%: INPUT
     "Minimum für Y: ";YL
1980 CX% = 40: &  POS (CX%,CY%): CALL CL%: INPUT "Maximum
     für Y: ";YH
1990 IF XL > XH THEN X = XL:XL = XH:XH = X
2000 IF YL > YH THEN X = YL:YL = YH:YH = X
2010 IF YL > 0 THEN YL = 0
2020 IF YH < 0 THEN YH = 0
2030 IF XL > 0 THEN XL = 0
2040 IF XH < 0 THEN XH = 0
2050 IF YL = YH AND XL = XH THEN  PRINT  CHR$
     (7);"Eingabefehler! Bitte wiederholen!": GOTO 1950
2060 IF YL = YH THEN  GOSUB 2420
2070 IF XL = XH THEN  GOSUB 2400
2080 VTAB 13: PRINT "Die Darstellungsgrenzen wurden wie
     folgt festgelegt: "
2090 &  POS (0,16): PRINT "Kleinstes X: ";XL;: CALL CL%
2100 &  POS (40,16): PRINT "Größtes X:  ";XH
2110 &  POS (0,18): PRINT "Kleinstes Y: ";YL;: CALL CL%
2120 &  POS (40,18): PRINT "Größtes Y:  ";YH;: CALL CL%
2130 PRINT : VTAB 22: PRINT "Ist das richtig? <J/N>";:
     CALL CL%
2140 GOSUB 3400: IF A$ = "J" THEN  PRINT : GOTO 2350
2150 I = 0:X0 = I:Y0 = I:X1 = I:Y1 = I:MX = I:MY = I:BX =
     I:BY = I: REM clr Hilfsvar.
2160 PRINT : VTAB 22: PRINT "Wollen Sie die
     Darstellungsgrenzen komplett neu eingeben? <J/N> ";:
     GOSUB 3400: IF A$ = "J" THEN  PRINT : VTAB 1: PRINT
     "Dann bitte Eingabe wiederholen!": GOTO 1950
2170 HOME : PRINT "Kleinere Manipulationen an den Achsen.
     Wollen Sie Erläuterungen? <J/N>";: GOSUB 3400: PRINT
2180 IF A$ = "J" THEN ID$ = "3": GOSUB 5510
2190 HOME : PRINT : VTAB 3: PRINT " x-Achse strecken?
     <J/N> ";: GOSUB 3400: IF A$ = "J" THEN  GOSUB 1700
2200 PRINT : VTAB 5: PRINT " y-Achse strecken? <J/N> ";:
     GOSUB 3400: IF A$ = "J" THEN  GOSUB 1750
2210 PRINT : VTAB 8: PRINT "Nun bitte das Verhältnis
     positiver/negativer Achsenabschnitt:": PRINT : PRINT "
     Für x-Achse? <J/N> ";: GOSUB 3400: IF A$ = "J" THEN
     GOSUB 1800
2220 PRINT : VTAB 12: PRINT " Für y-Achse? <J/N> ";: GOSUB
     3400: IF A$ = "J" THEN  GOSUB 1850
2230 IF (X1 OR X0) AND  NOT (X1 AND X0) OR (Y1 OR Y0) AND
     NOT (Y1 AND Y0) THEN  PRINT : PRINT "Nie mit 0
     strecken oder stauchen!" CHR$ (7): GOTO 2190
2240 VTAB 20: PRINT "Alles o.k.? <J/N> ";: GOSUB 3400: IF
     A$ = "N" THEN  PRINT : VTAB 2: PRINT "Bitte
     wiederholen!";: CALL CL%: GOTO 2190
2250 IF X0 AND X1 THEN XL = XL / X0 * X1:XH = XH / X0 * X1
2260 IF Y0 AND Y1 THEN YL = YL / Y0 * Y1:YH = YH / Y0 * Y1
2270 IF MX OR BX THEN X0 = XH - XL:X1 = MX + BX:XH = MX /
     X1 * X0:XL =  - BX / X1 * X0
2280 IF MY OR BY THEN Y0 = YH - YL:Y1 = MY + BY:YH = MY /
     Y1 * Y0:YL =  - BY / Y1 * Y0
2290 HOME : GOTO 2080
2330 :
2340 REM  Umrechnung Bildschirm
2350 MX = 559 / (XH - XL):BX =  - MX * XL
2360 MY =  - 191 / (YH - YL):BY =  - MY * YH
2370 DEF  FN XS(X) =  INT (MX * X + BX + .5): DEF  FN
     YS(X) =  INT (MY * X + BY + .5)
2380 RETURN
2390 REM x-Achse an y-Achse anpassen
2400 FA = 279 / 191:XL = FA * (YL - YH) / 2:XH = FA * (YH
     - YL) / 2: RETURN
2410 REM y-Achse an x-Achse anpassen
2420 FA = 279 / 191:YL = (XL - XH) / (2 * FA):YH = (XH -
     XL) / (2 * FA): RETURN
2430 :
2440 REM für Skalierung
2450 X0 =  INT ( FN LG(X)): IF X0 <  FN LG(X) THEN X0 = X0
     + 1
2460 ST = X0 - 1: IF  FN LG(X) - X0 <  FN LG(2) - 1 THEN
     ST = ST - 1
2470 X0 =  EXP ( LOG (10) * X0):ST =  EXP ( LOG (10) *
     ST): RETURN
2480 :
2580 REM Eingabe Parameter
2590 REM  Eingabe von T
2600 IF  RIGHT$ (P$,1) = "K" THEN 2670
```

Festplattenkomplettlösung für jedermann

Mit der Firma Frank & Britting GmbH, die auf Festplatten spezialisiert ist, konnten wir ein extrem günstiges Sonderangebot aushandeln, das eine Festplattenkomplettlösung selbst für Apple-Besitzer mit kleinerem Geldbeutel erschwinglich macht. Sie können unter zwei Varianten wählen:

Luxus-Lösung: 20-Megabyte-Festplatte MDB20 (MDB = Mobile Datenbox) + Megaboard-Controller + Handbuch + 3 Konfigurationsdisketten + Anschlußkabel + DB-Meister-Dateiverwaltungsprogramm + Handbuch + 2 Programmdisketten zum Gesamtpreis von nur DM 3199,- inkl. MwSt.
Standard-Lösung: 10-Megabyte-Festplatte MDB10 + Megaboard-Controller + Handbuch + 3 Konfigurationsdisketten + Anschlußkabel + DB-Meister-Dateiverwaltungsprogramm + Handbuch + 2 Programmdisketten zum Gesamtpreis von nur DM 2799,- inkl. MwSt.
(jeweils 6 Monate Garantie.)

Wozu eine Festplatte?

Für eine Festplatte sprechen zwei Argumente, nämlich erstens der bedeutend größere Datenspeicher und zweitens die sehr hohe Zugriffszeit.
● 80- und 160-Spur-Laufwerke haben zwar einen größeren Datenspeicher als die normalen 35-Spur-Laufwerke, doch ist die Zugriffszeit hier wie dort bescheiden.
● RAM-Karten haben zwar eine hohe Zugriffszeit, doch ist der Datenspeicher flüchtig: Strom aus – Daten weg!
Für unsere Festplatten-Standard-Lösung (MDB10) gilt demgegenüber:
– Die Nettospeicherkapazität beträgt hier ca. 9.650K = ca. 9.880.000 Zeichen = ca. 15 Disketten mit 160 Spuren = ca. 69 Disketten mit 35 Spuren.
– Die Datenübertragungsrate beträgt auf unterster Ebene ca. 44K/s = ca. 45.000 Zeichen/s. Zum Vergleich beläuft sich die Übertragungsrate bei normalen Diskettenlaufwerken auf ca. 7,5K/s und bei RAM-Karten auf ca. 44K/s, d.h. RAM-Karten sind also keineswegs schneller, sondern meist sogar langsamer als die MDB-Festplatte.
Sie erhalten deshalb mit unserer Festplattenlösung einen großen externen Massenspeicher bei sehr hoher Datenübertragungsrate zu einem äußerst niedrigen Preis.

Unkomplizierte Installation

Die eigentliche Festplatte, also die „Mobile Datenbox" MDB, ist ein kleines rechteckiges Kästchen mit der Länge einer Handfläche. Das eine Ende des Flachbandkabels wird in die MDB, das andere Ende in den Megaboard-Controller gesteckt, der wie eine normale Erweiterungskarte aussieht und in jeden Slot von 1-7 paßt. Die MDB hat ein eigenes Netzteil. Deshalb brauchen Sie jetzt nur noch die MDB und danach den Apple einzuschalten, und fertig ist die Installation. Im Gegensatz zur Megacore-Festplatte, die wegen des Apple-Netzteil-Wechsels von Apple-Händlern eingebaut werden sollte, ist die Installation der MDB von jedermann problemlos zu bewerkstelligen.

Problemlose Konfigurierung

Danach booten Sie die mitgelieferte Systemdiskette DOS-CONFIG von einem 35-Spur-Laufwerk, tippen „I" für Initialisieren, geben die Festplattenparameter ein (bei MDB10: 4 Köpfe, 306 Zylinder, Präkompensation ab dem 256. Zylinder und Schneller Zugriff Ja), und nach zwei weiteren Kontrollbestätigungen beginnt die Initialisierung. Dies dauert ca. 14 Minuten für die MDB10, und nunmehr kann die Festplatte mit den Betriebssystemen beschrieben werden.

Als Betriebssysteme können Sie im einzelnen verwenden:
– DOS 3.3,
– Diversi-DOS 2C und 4C,
– Apple-Pascal 1.1 und 1.2,
– ProDOS 1.0.1, 1.0.2, 1.1.1,
– CP/M 2.2 56K (z.Zt nur diese CP/M-Version).

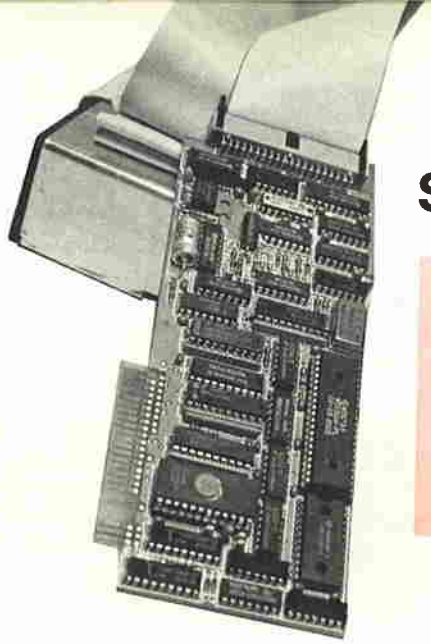
Die Konfigurierung, d.h. die Speicherverteilung auf die verschiedenen Betriebssysteme, sollte mit Bedacht überlegt werden. Nicht-benötigte Betriebssysteme können Sie natürlich weglassen. Nehmen wir an, Sie brauchen zunächst nur DOS 3.3. Wie wird nun konfiguriert?

Sie booten nach der obigen Initialisierung die Festplatte mit PR#7 und wählen aus dem automatisch erscheinenden Bildschirm-Menü „C" für „Configuration". Dann Tippen Sie „D" für DOS 3.3, danach „V" für Volumes, danach z.B. „10" für 10 Volumes = 2 x 10 x 140 = 2800K (1 Volume entspricht einem 35-Spur-Doppellaufwerk). Nun können Sie nach zwei weiteren Kontrollbestätigungen die neue Konfiguration auf die Diskette schreiben, was nur wenige Sekunden dauert. So einfach ist das! Später können Sie dann, sobald Sie mit der Festplatte besser vertraut sind, bei Bedarf die Konfigurationen für ProDOS, Apple-Pascal und CP/M nachholen.

Wie spricht man die MDB an?

Sie werden sich nun vielleicht fragen, wie Sie die Festplatte ansprechen müssen und ob Sie neue DOS-Befehle lernen müssen. Mitnichten! Booten Sie einfach die Festplatte mit PR#7 – übrigens heißt es immer PR#7, auch wenn Sie die Megaboard-Karte z.B. in Slot 4 gesteckt haben – und danach z.B.
INIT HELLO, S7, V3, D1
Damit wird das Volume in „Drive 1, Slot 7" formatiert. Nun können Sie mit
CATALOG
RUN usw.
auf dieses Volume wie üblich zugreifen.

er-Leser ·



Sonderangebot für Peeker-Leser

Wie schnell ist die MDB?

Die Datenübertragungsrate beträgt bei der MDB physikalisch 5 MBits/s. Dies ist ein theoretischer und niemals erreichbarer Wert. Auf Spur-Sektor- bzw. Blockebene beträgt die praktisch erzielbare Übertragungsrate ca. 44K/s bis 47,5K/s, je nach Betriebssystem. Einige anschauliche Beispiele: Auf Sektorebene läßt sich ein 140K-DOS-Volume mit dem Programm MDB.KOPY in 6,2s auf ein anderes 140K-Volume überspielen. Auf Dateiebene läßt sich eine 1-Megabyte-ProDOS-Datei mit dem Programm PROFID in 122s duplizieren. Dies sind umgerechnet 17.000 Bytes/s für Lesen bzw. Schreiben.

Was läuft mit der MDB?

Sie können normale Disk-II-Laufwerke und beispielsweise Erphi-160-Spur-Laufwerke im Mischbetrieb einsetzen. Die zusätzliche Verwendung von RAM-Karten ist bei der MDB überflüssig, da mit diesen in der Regel keine höheren Übertragungsraten erzielt werden können. Die Accelerator IIe funktioniert einwandfrei in Verbindung mit dem Megaboard-Controller (Slot-7-Schalter auf 1MHz einstellen). Die Speedemon funktioniert jedoch wegen des Cache-Speichers leider nicht.

DB-Meister

Speziell für die zukünftigen MDB-Besitzer wurde das Adreß- und Schemabriefprogramm DB-Meister, das in der normalen Diskettenversion DM 290,- kostet, zum Festplattenbetrieb umgeschrieben. Technische Daten:

– Datensatzlänge bis zu 230 Zeichen, aufteilbar in bis zu 25 Felder; maximal 1000 Datensätze pro Volume
– 2 variable Indexfelder und 1 zusätzliches, konstantes Suchfeld.
– Maskengenerator-Modul für die freie Definition von Bildschirmeingabe- und Ausdruck-Masken.
– Dateipflege-Modul zum Neueingeben, Ändern, Löschen, Ansehen, Bildschirm-Ausdruck usw.
– Sortier- und Filter-Module zum Sortieren und Untersortieren nach Indexfeldern und zum selektiven oder kumulierenden Filtern nach beliebigen Feldern.
– Druck-Modul zum Ausdrucken von Selbstklebe-Etiketten, tabellarischen Listen und Schemabriefen mit automatischem Einschub von Adressen und Anreden sowie weiteren Feldeinschüben im Briefkörper; Einzelblatt und Endlospapier verwendbar; Druckertreiber für Typenraddrucker mit anderer Typenradbelegung können eingebunden werden.
– Brief-Modul zum Schreiben der Schemabriefe; bei Bedarf separat verwendbar.
– Schnelles Backup-Programm (nur 10 Minuten für 2,5 Megabytes bzw. 18 DB-Meister-Dateien).
– Diverse Spezial-Module zur Indexfeld-Umdefinierung, zur Sortierung nach Kundennummern usw.

Alle Programm-Module laufen auf einem Apple IIe oder II+ mit MDB (Ausnahme: Das Brief-Modul funktioniert nur auf dem IIe mit 80-Zeichenkarte, doch können beliebige Fremdtexte eingelesen werden.).

Das DB-Meister-Programm eignet sich für normale Adreßverwaltung sowie für Dateien aller Art mit vielen, aber kleinen Datensätzen (max. 230 Zeichen). Das Programm ist nicht für das Schreiben von Rechnungen o.ä. gedacht.

Die Verarbeitungsgeschwindigkeit des „DB-Meisters" ist bei der MDB extrem schnell. Einige Sekundenangaben, bezogen auf eine Datei mit 500 Datensätzen zu je 220 Zeichen:

– Modul-Wechsel: 1s
– Suche nach Indexfelder: 0,3s
– Suche nach Nicht-Indexfeldern: 17s
– Filtern und Untersortieren: 25s
– Bildschirm-„Drucken": 38s

Noch ein etwas anschaulicheres Beispiel: Wenn Suchfelder neu vergeben werden sollen, muß die alte Indexdatei (ca. 50 Sektoren) und die ganze Hauptdatei (ca. 450 Sektoren) eingelesen werden und dann nach dem automatischen Heraussuchen der neuen Suchwörter die neue Indexdatei abgespeichert werden. Dieser normalerweise zeitraubende Prozeß dauert auf der MDB nur 14s.

Wie wird bestellt?

Sie senden Ihre Bestellung an den Hüthig-Software-Service. Sie erhalten dann von der Firma Frank & Britting eine Vorausrechnung, nach deren Überweisung Ihnen von dort die MDB10 bzw. MDB20, der Megaboard-Controller, das Handbuch und die Konfigurierungsdisketten mit 6 Monaten Garantie geliefert werden. Gleichzeitig erhalten Sie vom Hüthig-Software-Service das DB-Meister-Programm (2 Disketten und Handbuch) in der für die MDB bereits angepaßten Version. Nach einer geringfügigen Änderung im Hello-Programm können Sie diese Neuversion des DB-Meisters übrigens auch zusätzlich auf normalen 35-Spur-Laufwerken einsetzen.

Zur Bestellung können Sie eine der im Peeker eingehefteten Bestellkarten verwenden. Stichwort:

1 x MDB10-Sonderangebot für DM 2799,-
oder
1 x MDB20-Sonderangebot für DM 3199,-

Hüthig PUBLIKATION

SOFTWARE SERVICE
Im Weiher 10 · 6900 Heidelberg 1

```
2610 CY% = 3:CX% = 0: &  POS (CX%,CY%): INPUT "Bitte T
     eingeben: ";T
2620 VTAB 4: PRINT  SPC( 14);"T =  ";T: CALL CL%
2630 PRINT : PRINT  TAB( 20);"Richtig? <J/N> ";
2640 GOSUB 3400: IF A$ = "N" THEN 2610
2650 RETURN
2660 REM  T in Schleife
2670 CY% = 2:CX% = 0: &  POS (CX%,CY%): INPUT "Bitte
     Startwert: ";T0:CX% = 40: &  POS (CX%,CY%): INPUT
     "und Endwert für T eingeben: ";T1
2680 CX% = 0:CY% = 4: &  POS (CX%,CY%): INPUT "Nun bitte
     die Schrittweite eingeben: ";TS
2690 IF TS = 0 THEN  VTAB 7: PRINT  CHR$ (7);"Nie
     Schrittweite 0!": GOTO 2680
2700 IF TS < 0 THEN 2730
2710 IF T1 < T0 THEN T = T1:T1 = T0:T0 = T
2720 GOTO 2740
2730 IF T0 < T1 THEN T = T0:T0 = T1:T1 = T
2740 L1 = 1:L2 = 7: GOSUB 3450: VTAB 2
2750 PRINT "T          läuft von: ";T0
2760 PRINT "                  bis: ";T1
2770 PRINT "   mit Schrittweite: ";TS
2780 PRINT : PRINT  TAB( 10);"Ist das richtig? <J/N> ";:
     GOSUB 3400: IF A$ = "N" THEN L1 = 2:L2 = 6: GOSUB
     3450: GOTO 2670
2790 RETURN
2800 :
2810 REM  Eingabe von S
2820 CX% = 0:CY% = 8: &  POS (CX%,CY%): INPUT "Bitte S
     eingeben: ";S
2830 VTAB 9: PRINT  SPC( 15);"S = ";S: CALL CL%
2840 VTAB 11: PRINT  TAB( 20);"Richtig? <J/N> ";: GOSUB
     3400: IF A$ = "N" THEN 2820
2850 RETURN
2860 :
2870 REM  Eingabe von R
2880 CX% = 0:CY% = 14: &  POS (CX%,CY%): INPUT "Bitte R
     eingeben: ";R
2890 VTAB 15: PRINT  SPC( 15);"R = ";R: CALL CL%
2900 VTAB 17: PRINT  TAB( 20);"Richtig? <J/N> ";: GOSUB
     3400: IF A$ = "N" THEN 2880
2910 RETURN
2920 :
2940 REM Für Parameterliste
2950 FOR J = 1 TO I: &  POS ( FN P1(J),CY% +  FN P2(J)):
     PRINT P(K,I - J + 1): NEXT : RETURN
2960 :
2990 REM  für Parametertabelle
3000 N$ = "nicht verwendet"
3010 ON  VAL (P$) GOTO 3030,3040,3060,3050,3070,3080,3090
3020 S$ = N$:T$ = N$:R$ = N$: RETURN
3030 R$ =  STR$ (R):S$ = N$:T$ = N$: RETURN
3040 S$ =  STR$ (S):R$ = N$:T$ = N$: RETURN
3050 T$ =  STR$ (T):R$ = N$:S$ = N$: RETURN
3060 R$ =  STR$ (R):S$ =  STR$ (S):T$ = N$: RETURN
3070 R$ =  STR$ (R):T$ =  STR$ (T):S$ = N$: RETURN
3080 S$ =  STR$ (S):T$ =  STR$ (T):R$ = N$: RETURN
3090 R$ =  STR$ (R):S$ =  STR$ (S):T$ =  STR$ (T): RETURN
3100 :
3140 REM alle Parameter eingeben
3150 IF P$ = "0" THEN 3180
3160 HOME : PRINT  TAB( 10);"Parametereingabe"
3170 ON  VAL (P$) GOSUB 3190,3200,3220,3210,3230,3240,3250
3180 HOME : RETURN
3190 GOSUB 2880: RETURN
3200 GOSUB 2820: RETURN
3210 GOSUB 2600: RETURN
3220 GOSUB 2820: GOSUB 2880: RETURN
3230 GOSUB 2600: GOSUB 2880: RETURN
3240 GOSUB 2600: GOSUB 2820: RETURN
3250 GOSUB 2600: GOSUB 2820: GOSUB 2880: RETURN
3260 :
3290 REM  Kurzzusammenfassung
3300 HOME : PRINT "Die  Funktion lautet:": PRINT F$
3310 VTAB 4: PRINT "Darstellungsgrenzen:": &  POS (9,4):
     PRINT "Kleinstes X: ";XL;: &  POS (40,4): PRINT
     "Größtes X: ";XH
3320 &  POS (9,5): PRINT "Kleinstes Y: ";YL;: &  POS
     (40,5): PRINT "Größtes Y: ";YH
3330 VTAB 7: PRINT "Parameter: ": GOSUB 3000
3340 &  POS (0,7): PRINT "R:"R$: IF I > 0 AND R$ <  > N$
     THEN CY% = 7:K = 0: GOSUB 2950
3350 &  POS (0,11): PRINT "S:"S$: IF I > 0 AND S$ <  > N$
     THEN CY% = 11:K = 1: GOSUB 2950
3360 &  POS (0,15): PRINT "T:"T$: IF I > 0 AND T$ <  > N$
     THEN CY% = 15:K = 2: GOSUB 2950
3370 VTAB 21: PRINT "Abszisse X = ": RETURN
3380 :
3390 REM Teste ob ja oder nein
3400 GET A$: IF A$ <  >  CHR$ (3) AND A$ <  > "J" AND A$ <
     > "j" AND A$ <  > "N" AND A$ <  > "n" THEN 3400
3410 IF A$ > "Z" THEN A$ =  CHR$ ( ASC (A$) - 32)
3420 IF A$ <  >  CHR$ (3) THEN  RETURN
3430 POP : GOTO 6460
3440 REM  lösche zwischen L1 und L2 Bildschirm
3450 FOR L = L1 - 1 TO L2 - 1: &  POS (0,L): CALL CL%:
     NEXT : RETURN
3460 :
3480 REM  Zeichne Achsen
3490 REM  und skaliere
3500 IF XL = YH AND YL = XH THEN  RETURN
3510 &  HGR : &  HPLOT 0,0 TO 559,0 TO 559,191 TO 0,191 TO
     0,0: REM  generiere Rahmen
3520 &  HPLOT  FN XS(0),0 TO  FN XS(0),191: &  HPLOT 0, FN
     YS(0) TO 559, FN YS(0):K = 0: IF YL = 0 THEN K =  - 2
3530 IF YH = 0 THEN K = 2
3540 X = XH: IF  ABS (XL) > XH THEN X =  ABS (XL)
3550 GOSUB 2450: FOR X = ST TO X0 STEP ST
3560 IF X < XH THEN  &  HPLOT  FN XS(X), FN YS(0) + K - 2
     TO  FN XS(X), FN YS(0) + K + 2
3570 IF  - X > XL THEN  &  HPLOT  FN XS( - X), FN YS(0) +
     K - 2 TO  FN XS( - X), FN YS(0) + K + 2
3580 NEXT :K = 0: IF XL = 0 THEN K = 2
3590 IF XH = 0 THEN K =  - 2
3600 X = YH: IF  ABS (YL) > X THEN X =  ABS (YL): GOSUB
     2450
3610 GOSUB 2450: FOR X = ST TO X0 STEP ST
3620 IF X < YH THEN  &  HPLOT  FN XS(0) + K - 2, FN YS(X)
     TO  FN XS(0) + K + 2, FN YS(X)
3630 IF  - X > YL THEN  &  HPLOT  FN XS(0) + K - 2, FN YS(
     - X) TO  FN XS(0) + K + 2, FN YS( - X)
3640 NEXT : RETURN
3650 :
3690 REM Funktionseingabe
3700 I = 0: HOME : PRINT "Funktionseingabe: Wünschen Sie
     Erläuterungen? <J/N>";: GOSUB 3400
3710 IF A$ = "J" THEN ID$ = "4": GOSUB 5510
3720 HOME : PRINT "Bitte geben Sie die darzustellende
     Funktion ein!": ONERR  GOTO 3900
3730 L1 = 10:L2 = 15: GOSUB 3450: &  POS (0,9): PRINT F$:
     VTAB 10
3740 &  FN ,F$:F$ = "Y=" + F$:P$ =  STR$ ( PEEK (255))
3750 R = 1.4:S = 2.1:T = 3.31: REM  Zufallswerte Parameter
3760 I = 0:Y =  FN Y(1.1084): REM  Syntax-Test
3770 R = 0:T = 0:S = 0: IF P$ < "4" THEN 3810
3780 VTAB 13: PRINT "Es wurde der Parameter T verwendet."
3790 PRINT "Soll T in einer Schleife mit fester
     Schrittweite durchlaufen werden? <J/N>";
3800 GOSUB 3400: IF A$ = "J" THEN P$ = P$ + "K"
3810 &  POS (0,9): CALL CL%: PRINT F$;: CALL CL%: &  POS
     (0,21): PRINT "Ist das richtig? <J/N> ";: CALL CL%
3820 GOSUB 3400: IF A$ = "N" THEN  PRINT "Bitte
     korrigieren!": GOTO 3730
3830 ONERR  GOTO 1300: REM wieder zurück auf normale
     ONERR-Routine
3840 RETURN
3880 :
3890 REM Spezial ONERR-Routine
3900 ER =  PEEK (222): CALL 3087: IF ER <  > 16 THEN 3770
3910 VTAB 17: PRINT  CHR$ (7)"Fehler bei der
     Funktionseingabe! Bitte auf korrekte Syntax achten!
     Nochmal!"
3920 GOTO 3730
3930 :
3940 REM  SUPERDUMP STARTEN
3950 PRINT : POKE 104,8: PRINT  CHR$ (4)"RUN
     SUPERDUMP.EPSON"
3960 :
4980 REM ═══════════════════════════════════
4990 REM  Disk-Menü
5000 POKE PG%,1: TEXT : HOME : PRINT "Disk-Menü:
     (Betätigen Sie  0 ... 6, Leertaste zeigt Grafik!)":
     ONERR  GOTO 5700
5010 VTAB 5: PRINT " 1 - Inhaltsverzeichnis der Diskette"
5020 PRINT : PRINT " 2 - Grafik von Disk laden"
5030 PRINT : PRINT " 3 - Grafik auf Disk speichern"
5040 PRINT : PRINT " 4 - aktives Laufwerk auswählen"
5050 PRINT : PRINT " 5 - Erläuterungen abrufen"
5060 PRINT : PRINT " 6 - Superdump"
5070 VTAB 20: PRINT " 0 - Zurück ins Hauptmenü"
5080 GET A$: IF A$ = " " THEN  GOSUB 1510: GOTO 5080
5090 IF A$ < "0" OR A$ > "6" THEN 5080
5100 ON  VAL (A$) GOTO 5140,5170,5220,5380,5270,3950
5110 ONERR  GOTO 1300
5120 GOTO 6460: REM zurück ins Hauptmenü
5130 REM CATALOG
5140 PRINT : HOME : PRINT D$"CATALOG,D"DR$",S"SL$
5150 PRINT " Bitte beliebige Taste betätigen ";: GET A$:
     GOTO 5000
5160 REM Neues Bild laden
5170 TEXT : HOME : POKE PG%,1
5180 INPUT "File-Sammelname ";A$
```

```
5190 POKE  - 16299,0: PRINT D$"BLOAD"A$".A,S"SL$",D"DR$
5200 POKE  - 16300,0: PRINT D$"BLOAD"A$".M": GOTO 5000
5210 REM Grafik auf Disk speichern
5220 POKE PG%,1: TEXT : HOME
5230 INPUT "File-Sammelname ";A$
5240 POKE  - 16299,0: PRINT
     D$"BSAVE"A$".A,A$2000,L$2000,S"SL$",D"DR$
5250 POKE  - 16300,0: PRINT D$"BSAVE"A$".M,A$2000,L$2000":
     GOTO 5000
5260 REM Erläuterungen
5270 POKE PG%,1: TEXT : HOME : PRINT "Welche Erläuterungen
     wünschen Sie? (Betätigen Sie 0 ... 5!)"
5280 VTAB 5: PRINT " 1 - Allgemeine Erläuterungen"
5290 PRINT : PRINT " 2 - Erläuterungen zu
     Darstellungsgrenzen"
5300 PRINT : PRINT " 3 - Erläuterungen zu
     Achsenmanipulationen"
5310 PRINT : PRINT " 4 - Erläuterungen zur
     Funktionseingabe"
5320 PRINT : PRINT " 5 - Erläuterungen zum Disk-Menü"
5330 VTAB 20: PRINT " 0 - zurück ins Disk-Menü"
5340 GET A$: IF A$ < "0" OR A$ > "5" THEN 5340
5350 IF A$ = "0" THEN5 5000
5360 ID$ = A$: GOSUB 5510: GOTO 5270
5370 REM Aktives Disk-Laufwerk wählen
5380 TEXT : HOME : POKE PG%,1: PRINT "Momentan aktiv:"
5390 PRINT "          Slot: ";SL$
5400 PRINT "         Drive: ";DR$
5410 PRINT : PRINT "Bitte auswählen:"
5420 PRINT : PRINT "Slot (1 bis 7)   ";
5430 GET A$: IF A$ < "1" OR A$ > "7" THEN 5430
5440 SL$ = A$: PRINT SL$
5450 PRINT : PRINT "Drive (1 oder 2) ";
5460 GET A$: IF A$ <  > "1" AND A$ <  > "2" THEN 5460
5470 DR$ = A$: PRINT DR$
5480 PRINT : PRINT " Beliebige Taste betätigen! ";: GET
     A$: GOTO 5000
5490 :
5500 REM  Help-File ausgeben
5510 PRINT : HOME : PRINT D$"OPEN PLOT.HELP."ID$:X = 0
5520 PRINT D$"READ PLOT.HELP."ID$
5530 INPUT A$: IF A$ = ".EOF" THEN 5560
5540 PRINT A$:X = X + 1: IF X = 16 THEN  GOSUB 5570:X = 0
5550 GOTO 5530
5560 PRINT D$"CLOSE": PRINT : GOSUB 5570: PRINT : RETURN
5570 POKE  - 16368,0: PRINT : PRINT "      Bitte beliebige
     Taste betätigen!": PRINT
5580 IF  PEEK (KE%) < 128 THEN 5580
5590 IF A$ <  > ".EOF" THEN  &  POS (0, PEEK (37) - 3):
     CALL  - 958
5600 POKE  - 16368,0: RETURN
5610 :
5690 REM Disk-ONERR-Routine
5700 POKE  - 16300,0:ER =  PEEK (222): IF  NOT ER OR ER >
     15 THEN  PRINT  CHR$ (7) CHR$ (21)"Fehler Code #
     ";ER: END
5710 IF ER = 6 THEN  PRINT "Der gesuchte File existiert
     nicht auf dieser Diskette!": GOTO 5800
5720 IF ER = 8 THEN  PRINT "Schreib-Lesefehler! Diskette
     defekt?": GOTO 5800
5730 IF ER = 9 THEN  PRINT "Auf dieser Diskette ist kein
     Platz mehr. Wechseln Sie sie bitte aus!": GOTO 5800
5740 IF ER = 13 THEN  PRINT "Dieser File existiert schon
     und ist kein Binärfile! Vorsicht": GOTO 5800
5750 IF ER = 10 THEN  PRINT "Dieser File existiert schon
     und ist schreibgeschützt!": GOTO 5800
5760 IF ER = 4 THEN  PRINT "Diskette ist schreibgeschützt!
     Peeker Sammeldiskette?": GOTO 5800
5770 IF ER = 11 THEN  PRINT "Illegaler Filename!": GOTO
     5800
5780 IF ER = 5 THEN  PRINT "Bitte legen Sie eine Diskette
     mit File 'PLOT.HELP."ID$"' ein!": GOTO 5 800
5790 PRINT  CHR$ (7) CHR$ (21) CHR$ (13)"DOS-Fehler #";ER;
     CHR$ (7): END
5800 CALL 3087: PRINT  CHR$ (7): GET A$: PRINT : GOTO 5000
5810 :
5980 REM ═══════════════════════════════════
5990 REM  Initialisierende Schritte
6000 TEXT : HOME : PRINT  TAB( 16)"PLOT.3.E": IF  PEEK
     (104) < 64 THEN 7000
6010 ONERR  GOTO 1300: REM  Seele des Programms
6020 X = 0:Y = 0:X0 = 0:Y0 = 0:X1 = 0:Y1 = 0: DEF  FN Y(X)
     = 0:R = 0:T = 0:S = 0: REM zuerst
     geschwindigkeitsrelevante Variablen
6030 REM dann diverse Konstanten
6040 PI =  ATN (1) * 4:E =  EXP (1):PG% = 254:CL% =  -
     868:KE% =  - 16384
6050 D$ =  CHR$ (4): DIM P(2,20): DEF  FN LG(X) =  LOG (X)
     /  LOG (10):DR$ = "2":SL$ = "3": REM  Defaultwerte
     für Slot/Drive
6060 IF  PEEK (1014) <  > 101 OR  PEEK (1015) <  > 12 OR
     PEEK (2816) <  > 72 OR  PEEK (3087) <  > 104 OR  PEEK
     (4172) <  > 1 THEN  POKE 3087,96: PRINT
     D$"BRUNPLOT.BX"
6070 DEF  FN P1(I) = 80 * (I / 5 -  INT (I / 5)): DEF  FN
     P2(I) =  INT (I / 5)
6080 PRINT D$"PR#3": &  HGR : POKE PG%,0: GOSUB 1510: HOME
     : PRINT  TAB( 36)"PLOT.3.E"
6090 &  POS (30,9): PRINT "Von Hans-Martin Eng"
6100 &  POS (34,11): PRINT "Hirschberg,"
6110 &  POS (32,13): PRINT "September 1983,"
6120 &  POS (32,15): PRINT "November  1984,"
6130 &  POS (32,17): PRINT "Februar   1986"
6140 &  POS (30,21): GET A$
6150 HOME : PRINT "Wünschen Sie eine Kurzeinführung?
     Betätigen Sie Bitte 'J' oder 'N'.";
6160 GOSUB 3400: IF A$ = "N" THEN 6180
6170 ID$ = "1": GOSUB 5510
6180 GOSUB 3700: GOSUB 1900
6190 REM  Zusammenfassung
6200 HOME : PRINT "Zusammenfassung:"
6210 PRINT "Die Funktion lautet:": PRINT : PRINT F$
6220 PRINT : PRINT : PRINT "Parameter: ";: ON  VAL (P$)
     GOTO 6240,6250,6270,6260,6280,6290,6300
6230 PRINT : PRINT "Es werden keine Parameter verwendet.":
     GOTO 6320
6240 PRINT "R": GOTO 6320
6250 PRINT "S": GOTO 6320
6260 PRINT "T";: GOTO 6310
6270 PRINT "R und S": GOTO 6320
6280 PRINT "R und T";: GOTO 6310
6290 PRINT "S und T";: GOTO 6310
6300 PRINT "R,S und T";
6310 IF  RIGHT$ (P$,1) = "K" THEN  PRINT " / T in
     Schleife"
6320 PRINT : PRINT : PRINT "Grenzen:": PRINT  TAB( 5);"X:
     ";XL;" ... ";XH
6330 PRINT  TAB( 5);"Y: ";YL;" ... ";YH
6340 VTAB 21: PRINT "Sind alle Eingaben richtig? <J/N> ";:
     GOSUB 3400: IF A$ = "J" THEN 6430
6350 HOME : PRINT "Wo traten noch Fehler auf?"
6360 &  POS (9,5): PRINT "1 - Funktion"
6370 &  POS (9,9): PRINT "2 - Darstellungsgrenzen"
6380 &  POS (9,15): PRINT "0 - Keine Fehler"
6400 &  POS (12,22): PRINT "Betreffende Taste drücken ";
6410 GET A$: IF A$ < "0" OR A$ > "2" THEN 6410
6420 ON  VAL (A$) GOSUB 3700,1900: GOTO 6200
6430 GOSUB 3160: REM  Parametereingabe
6440 POKE PG%,0: GOSUB 3500: GOSUB 1030: REM Zeichnen
6450 REM  entfällt
6460 POKE PG%,0: GOSUB 1500: HOME : PRINT  TAB(
     7)"Hauptmenü  -  Bitte wählen Sie aus :      <
     Betätigen Sie 0 ... 7 >"
6470 PRINT  TAB( 4)"< Leertaste zeigt wieder die Grafik,
     nochmals Leertaste für dieses Menü >": PRINT
     "═══════════════════════════════════
     ═══════════════════════════"
6480 &  POS (16,05): PRINT "1 - Darstellungsgrenzen
     ändern"
6490 &  POS (16,07): PRINT "2 - Parameter ändern"
6500 &  POS (16,09): PRINT "3 - Funktion ändern"
6510 &  POS (16,11): PRINT "4 - Bildschirm säubern"
6520 &  POS (16,13): PRINT "5 - Alte Daten auflisten"
6530 &  POS (16,15): PRINT "6 - Wieder zeichnen"
6540 &  POS (16,17): PRINT "7 - Disk-Menü"
6550 &  POS (16,22): PRINT "0 - aufhören"
6560 GET A$: IF A$ = " " THEN  GOSUB 1510: GOTO 6560
6570 IF A$ < "0" OR A$ > "7" THEN 6560
6580 ON  VAL (A$) GOTO 6660,6700,6740,6780,6820,6860,5000
6590 :
6600 REM  0 Programmende
6610 &  TEXT : PRINT  CHR$ (21): HOME : VTAB 10: PRINT
     "Hoffe, es hat Spaß gemacht !"
6620 VTAB 20: HTAB 36: PRINT "HME";
6630 END : CLEAR : GOTO 6000: REM für evtl. CONT
6640 :
6650 REM  1 Darstellungsgrenzen
6660 POKE PG%,0: GOSUB 1510: GOSUB 1900: GOSUB 3500: GOTO
     6460
6670 REM  Eingabe Grenzen/zeichne Achsen und skaliere/Menü
6680 :
6690 REM  2 Parametereingabe
6700 POKE PG%,0: GOSUB 1510: GOSUB 3150: GOTO 6460
6710 REM  Eingabe Parameter/Menü
6720 :
6730 REM  3 Funktionseingabe
6740 POKE PG%,0: GOSUB 1510: GOSUB 3700: GOSUB 3160: GOTO
     6460
6750 REM  Eingabe
     Funktion/Parameter?/Parametereingabe/Menü
```

```
6760 :
6770 REM  4 Bildschirm
6780 I = 0: POKE  - 16302,0: GOSUB 3500: GOTO 6460
6790 REM  Zeichne Achsen und Skala/Menü
6800 :
6810 REM  5 Datenliste zeigen
6820 POKE PG%,0: GOSUB 1510: GOSUB 3300: GET A$: GOTO 6460
6830 REM  Zusammenfassung / Menü
6840 :
6850 REM  6 Zeichnen
6860 POKE PG%,1: GOSUB 1500: GOSUB 1020: GOTO 6460
6870 REM  Grafikseite/Menü
6880 :
6960 REM  Pointer in ($67) falsch gesetzt, das Programm
     liegt in der
6970 REM  1. Grafikseite. EXEC-File PLOT.3.PRO korrigiert
     das, lädt
6980 REM  und startet PLOT.3.E von Neuem
6990 :
7000 VTAB 8: HTAB 5: PRINT "Programm PLOT.3.E wird
     geladen": PRINT : HTAB 14: PRINT "Bitte warten"
7010 PRINT  CHR$ (4)"EXEC PLOT.3.PRO"
7020 :
7100 REM **********************
7120 REM *    P L O T  3.E    *
7140 REM * für APPLE //e, //c *
7160 REM *  Hans-Martin Eng   *
7180 REM *  24. Februar 1986  *
7200 REM **********************
```

PLOT.3.PRO (Exec-File)

```
POKE 50, 128
POKE 103, 1
POKE 104, 64
POKE 16384, 0
LOAD PLOT.3.E
CALL 54514
POKE 50,255
RUN
```

PLOT.PATCH

Ändern Sie die betreffenden Zeilen im Hauptprogramm wie folgt ab, wenn Ihr Druckerinterface doppelt hochauflösende Grafik ausdrucken kann.
Die Steuerzeichen sind natürlich druckerspezifisch. Besitzen Sie einen anderen Drucker, hilft vielleicht ein Blick ins Handbuch.

```
3940 REM  Ausdrucken
3950 PRINT  CHR$ (21): PRINT D$"PR#1": REM 80-Zeichen aus,
     Drucker ein
3960 PRINT  CHR$ (9) CHR$ (16);: REM  stelle
     Interface-Command-Charakter um
3970 REM  FX-80 spezifische Codes
3980 PRINT  CHR$ (27)"M" CHR$ (27)"1" CHR$ (8);: REM
     Elite-Zeichensatz, Linker Rand Spalte 8
3990 PRINT  CHR$ (27)"Q" CHR$ (90) CHR$ (27)"N" CHR$ (6);:
     REM  Rechter Rand Sp 90, unterer Rand
4000 PRINT  CHR$ (27)"D" CHR$ (8) CHR$ (24) CHR$ (40) CHR$
     (56) CHR$ (72) CHR$ (0): REM  Definiere
     Tabulatorstops
4010 PRINT  TAB( 43) CHR$ (27)"-1 PLOT.3.E " CHR$
     (27)"-0": REM  Kopfzeile unterstrichen
4020 PRINT  CHR$ (13) CHR$ (10) CHR$ (10)"Die Funktion
     lautet:"
4030 PRINT "F(X) = " MID$ (F$,3): PRINT
4040 PRINT  CHR$ (10) CHR$ (10)"Darstellungsgrenzen:" CHR$
     (9)"x läuft von:" CHR$ (9);
4050 PRINT XL CHR$ (9)"bis: "XH
4060 PRINT  CHR$ (9) CHR$ (9)"y läuft von:" CHR$ (9)YL
     CHR$ (9)"bis: "YH
4070 IF R$ = N$ AND T$ = N$ AND S$ = N$ THEN  PRINT :
     PRINT "Es werden keine Scharparameter verwendet.":
     GOTO 4150
4080 PRINT  CHR$ (13) CHR$ (10)"Parameterliste:": PRINT
     "Die Funktion wurde mit folgenden Parameterwerten
     gezeichnet:
4090 PRINT : IF R$ = N$ THEN 4110
4100 PRINT "  R:  ";:K = 0: GOSUB 4300: PRINT
4110 IF S$ = N$ THEN 4130
4120 PRINT : PRINT "  S:  ";:K = 1: GOSUB 4300: PRINT
4130 IF T$ = N$ THEN 4150
4140 PRINT : PRINT "  T:  ";:K = 2: GOSUB 4300: PRINT
4150 PRINT  CHR$ (13) CHR$ (10) CHR$ (10)"Hier die
     Grafik:"
4160 PRINT  CHR$ (27)"1" CHR$ (0)
4170 PRINT  CHR$ (16)"GZ": REM  kritischer Punkt. Wenn Ihr
     Grafikinterface keinen
4180 REM  Dumpbefehl für Double-Hires-Grafik kennt, müssen
     Sie auf
4190 REM  SUPERDUMP ausweichen
4200 PRINT : PRINT : PRINT  TAB( 80)"HME"
4210 PRINT D$"PR#0": PRINT D$"PR#3"
4220 GOTO 5000
4230 REM
4240 :
4290 REM  Parametertabelle
4300 FOR J = 1 TO I: PRINT  CHR$ (9);P(K,J);
4310 IF J / 5 =  INT (J / 5) THEN  PRINT
4320 NEXT J: PRINT
4340 RETURN
4350 :
5060 PRINT : PRINT " 6 - Ausdrucken"
6080 PRINT D$"PR#3": &  HGR : POKE PG%,0: GOSUB 1510: HOME
     : PRINT  TAB( 29)"PLOT.3.E / Direktdruck"
```

SUPERDUMP.PATCH

Damit Superdump die PLOT.3.E-Grafikfiles einlesen und ausdrucken kann, müssen einige Befehle eingefügt werden. Darum mußte an anderer Stelle Speicherplatz eingespart werden. Ändern Sie darum bitte im Applesoft-BASIC-Programm SUPERDUMP .EPSON oder SUPERDUMP.IMAGEWRITER die folgenden Zeilen ab.

```
1260 TEXT :D$ =  CHR$ (4): PRINT  CHR$ (12): HOME
1320 PRINT : PRINT D$"BRUN SUPERDUMP"
1325 IF  PEEK (64435) = 6 AND  PEEK (64448) = 0 THEN  POKE
     34499,32
1530 P1 =  PEEK (43634): REM ADDRESSE PARAMETERBLOCK
1540 P2 =  PEEK (43635):PB = P1 + 256 * P2
1740 INPUT "Welche Graphik laden ? ";F$
1750 IF F$ = "" THEN F$ = "BILD"
1755 PRINT "Plot.3.E - Grafikfile? <j/n>";: GOSUB 2150
1760 HGR : HOME : VTAB 22
1770 IF A$ = "J" THEN  PRINT D$"BLOAD"F$;".A,A$4000":F$ =
     F$ + ".M"
1780 PRINT D$"BLOAD";F$
1790 IF D = 1 THEN  POKE CO,2: & : REM   HGR 2 -> AUXRAM
1800 POKE  - 16300,0: POKE  - 16301,0: POKE  - 16297,0:
     POKE  - 16304,0
1920 POKE  - 16299,0: POKE  - 16302,0
1970 POKE  - 16300,0: POKE  - 16301,0
2040 POKE  - 16297,0: POKE  - 16304,0
2060 POKE CO,0: POKE HI,D
2080 & : REM  DRUCKE
2130 IF A$ = "N" THEN  HOME : PRINT D$"EXEC PLOT.3.PRO":
     END
```


Telefonische Bestellungen?

Da unsere Peeker-Disketten in offener Rechnung und nicht in dem für Sie teuren Nachnahme-Verfahren ausgeliefert werden, haben Sie bitte Verständnis dafür, daß wir **nur noch schriftliche Bestellungen annehmen.**

Sie können dazu beispielsweise die in jedem Peeker eingehefteten Bestellkarten verwenden.

Hüthig Software Service

PLOT.BX (Teil-Quellcode)

```
516 ******************************
517 * Linie beliebiger Steigung:
518 * Zuerst Differenz von y1 und y2.
519 *
520 LINE     SEC
521          LDA  Y2
522          SBC  Y1
523          STA  DY
524          BCS  Y1<Y2
525 *
526 * Absolutbetrag der Differenz,
527 *
528 Y1>Y2    SEC
529          LDA  #Ø
530          SBC  DY
531          STA  DY
532 * negative Y-Richtung.
533          LDA  #$FF
534          STA  YADD
535          BMI  WEITER      ; immer
536 *
537 * Positive Y-Richtung:
538 *
540 Y1<Y2    LDA  #1
541          STA  YADD
542 *
543 * Weiter mit den X-Koordinaten:
544 * Erst Differenz bestimmen,
545 *
546 WEITER   SEC
547          LDA  X2L
548          SBC  X1L
549          STA  DXL
550          LDA  X2H
551          SBC  X1H
552          STA  DXH
553          BCS  X1<X2
554 *
555 * Betragsbildung,
556 *
557 X1>X2    SEC
558          LDA  #Ø
559          SBC  DXL
560          STA  DXL
561          LDA  #Ø
562          SBC  DXH
563          STA  DXH
564 * negative X-Richtung.
565          LDA  #$FF
566          STA  XADDL
567          STA  XADDH
568          BMI  PARA?       ;immer
569 *
570 * Positive X-Richtung:
571 X1<X2    LDA  #Ø
572          STA  XADDH
573          LDA  #1
574          STA  XADDL
575 *
576 * Jetzt testen wir, ob die Gerade
577 * parallel zur X- oder Y-Achse.
578 *
579 PARA?    LDA  DY
580          BEQ  XLINE
581          LDA  DXL
582          BNE  TEST2
583          LDA  DXH
584          BEQ  YLINE
585 ********************************
586 * Kleine Modifikation des Trep-*
587 * penalgorithmus. Bringt       *
588 * besseres Bild.               *
589 ********************************
590 TEST2    SEC
591          LDA  DXH
592          BNE  TEST3
593          LDA  DXL
594          CMP  DY
595 TEST3    ROR
596          STA  FLAG
597 *
599 * Initialisiere Saldoreg. mit Ø
600 *
601 NOTPARA  SEC
602          LDA  #Ø
603          STA  DL
604          STA  DH
605 * und Schrittzähler:
606 * (STCNT = Ø - dY - dX)
607 *
608          SBC  DY
609 *
610          SEC
611          SBC  DXL
612          TAX              ;Low
613          LDA  #$FF
614          SBC  DXH
615          STA  STCNT       ;High
616 *
617 * Hier weiter für alle Geraden
618 * Ist D < Ø, dann in X-Richtung
619 * Ist D > Ø, dann Y-Richtung.
620 * Ist D = Ø, dann in Richtung der
621 * größeren Koordinatendifferenz.
622 *
623 NOTREADY LDA  DH
624          BMI  XSTEP
625          BNE  YSTEP
626          LDA  DL
627          BNE  YSTEP
628          BIT  FLAG
629          BPL  XSTEP
630 *
631 * Schritt in Y-Richtung:
632 * ----------------------
633 *
634 YSTEP    CLC
635 * neue Y-Koordinate
636          LDA  Y1
637          ADC  YADD
638          STA  Y1
639 * als Index
640          TAY
641 *
642          LDA  HGRADRL,Y
643          STA  IND1
644          LDA  HGRADRH,Y
645          STA  IND1+1
646 * Aktualisiere Saldoregister
647 * D.neu = D.alt - dX
648 *
649          SEC
650          LDA  DL
651          SBC  DXL
652          STA  DL
653          LDA  DH
654          SBC  DXH
655          STA  DH
656 *
657          INX
658          BNE  STINCY
659          INC  STCNT
660          BEQ  COMMON
661 *
662 STINCY   BIT  FLAG
663          BMI  XSTEP
664 *
665 * Ab hier XSTEP und YSTEP
666 * wieder gemeinsam.
667 *
668 COMMON   LDA  QUOTIENT
669 * AUXRAM oder MAINRAM ?
670          LSR
671          BCC  LPAGE2
672 *
673 LPAGE1   LDY  $CØ54
674          BCS  LPAGER
675 *
676 LPAGE2   LDY  $CØ55
677 *
678 LPAGER   TAY
679          LDA  (IND1),Y
680          ORA  BITBYTE
681          STA  (IND1),Y
682 *
683 * Schrittzähler testen.
684 *
685          BIT  STCNT
686          BMI  NOTREADY
687 *
688 * Schrittzähler >= Ø, die Gerade
689 * ist fertig .
690 *
691 READY    STA  $CØ54       ;MAINRAM
692 * Alte Endkoordinaten ggf. als
693 * neue Anfangskoordinaten.
694          LDA  X2H
695          STA  X1H
696          LDA  X2L
697          STA  X1L
698 * War mir ein Vergnügen.
699          RTS
700 *
701 *=========================
702 *
703 * Schritt in X-Richtung :
704 * -----------------------
705 *
706 * Statt Division Manipulation von
707 * Divisionsrest und Quotient aus
708 * vorangegangener Berechnung.
709 *
710 XSTEP     CLC
711          LDA  RESTL
712          ADC  XADDL
713 * Rest < Ø, Quotient erniedrigen.
714          BMI  XS1
715          CMP  #7
716          BNE  XS2
717 * Rest > 6, Quotient erhöhen
718          INC  QUOTIENT
719          LDA  #Ø
720          STA  RESTL
721          BEQ  XPAGE
722 *
723 XS1      DEC  QUOTIENT
724          LDA  #6
725          STA  RESTL
726          BNE  XPAGE
727 *
728 XS2      STA  RESTL
729 *
730 * Jetzt Bit-Position auswählen.
731 *
732 XPAGE    LDY  RESTL
733          LDA  BITTABLE,Y
734          STA  BITBYTE
735 * Schrittzähler erhöhen
736          INX
737          BNE  STINCX
738          INC  STCNT       ;letzter
739          BEQ  COMMON      ;Punkt
740 * Saldoregister:
741 * (D.neu = D.alt + dY)
742 STINCX   LDA  DL
743          CLC
744          ADC  DY
745          STA  DL
746 * Nun noch Punkt setzen.
747          BCC  COMMN1
748          INC  DH
749 * Doppelschritt?
750 COMMN1   BIT  FLAG
751          BMI  COMMON
752          JMP  YSTEP
```


Ältere Peeker-Hefte

können für DM 6,50 pro Heft zuzüglich Versandspesen angefordert werden. Vergriffene Hefte sind als Photokopien für DM 10,– pro Heft erhältlich. Mindestbestellmenge 2 Hefte.

Dr. A. Hüthig Verlag · Heidelberg

EDIT

Übungen zum Disk-Editor

Hinweis: Die Teile 2 und 3 mit dem kommentierten Quelltext des Editors erscheinen in den Heften 7/86 und 8/86. Nach dem Teil 1 in Heft 5/86 soll zunächst die praktische Anwendung des Editors anhand einfacher Übungsbeispiele demonstriert werden.

1. Zunächst erstellen wir ein Duplikat der ProDOS-Systemdiskette, die in der englischen Version USERS.DISK bzw. in der deutschen Version BENUTZER.DISK heißt.

2. Dann kopieren wir mit CONVERT, das sich auf der BENUTZER.DISK befindet (oder ersatzweise mit DOSTOPRO) von der Peeker-Sammeldisk #18 das Programm EDIT auf das Duplikat der BENUTZER.DISK.

3. Schließlich setzen wir noch, nachdem wir CONVERT verlassen haben, das Präfix mit
PREFIX/BENUTZER.DISK
auf unsere Übungsdiskette. Es ist wichtig, daß Sie die ersten Schritte mit EDIT auf einer Duplikatdiskette unternehmen, weil Sie dann keine Angst zu haben brauchen, daß Sie wertvolle Daten zerstören.

4. Jetzt starten wir den Editor mit
BRUN EDIT
Er meldet sich mit einer Bildschirmübersicht, die im Falle einer 80-Zeichenkarte wie Abb. 1 aussieht, andernfalls wie Abb. 2. Der Cursor ist auf der untersten Bildschirmzeile sichtbar, und alle nachfolgenden Befehle schließen wir mit Return ab.

5. Wir geben als erstes
PR/BENUTZER.DISK
ein, womit das Präfix auf BENUTZER.DISK eingestellt wird. Alternativ können wir auch mit
RP6,1
Slot 6 und Drive 1 festlegen. Würde sich die BENUTZER.DISK in Slot 4, Drive 2 befinden, so würde der Befehl
RP4,2
lauten.

6. Als nächstes tippen wir
A
womit der Puffer in A(scii) angezeigt ist. Wir sehen
000 - ... usw.
d.h. der Puffer ist im Moment noch leer. Sodann tippen wir
D
womit der Puffer als Dump (= Hex-Dump + Ascii) am Bildschirm ausgegeben wird, und zwar etwa so
000 - 00 00 00 usw.
weil der Puffer immer noch leer ist.

7. Nun geben wir
R0
oder bei Bedarf präziser
R0000
ein, womit der Block 0 (= $0000) von der Diskette in den Puffer gelesen wird (R = Read). Anschließend tippen wir erst
A
und dann
D
um die verschiedenen Anzeigeformen kennenzulernen.

8. Mit
R+
wird der nächste Block, also Block 1, in den Speicher gelesen, mit
R-
der vorangehende Block, also wieder Block 0.
Nun lesen wir mit
R0117
oder kürzer
R117
den letzten Block der 280-Block-Diskette, also Block $0117 bzw. dezimal 279, ein. Dieser Block ist auf der BENUTZER.DISK noch leer, und die Anzeige entspricht so der des leeren Puffers beim Programmstart.

9. Wir geben jetzt
S000'MEIN NAME IST HASE
ein, womit ab der relativen Byte-Position (= Offset) $000 der angegebene String in den Puffer geschrieben wird. Da der Puffer ebenso wie ein ProDOS-Block $200 Bytes umfaßt, können wir mit der S-Option einen String ab S000 bis S1FF einfügen. Wir beachten ferner den Apostroph vor „'MEIN...", der besagt, daß der String mit Bit 7 off in den Puffer übernommen wird. Um dies zu überprüfen, tippen wir, wie wir jetzt bereits gewohnt sind,
A
und danach
D
wobei wir im letzteren Fall Hex- und Ascii-Darstellung gleichzeitig vorfinden.
Nun geben wir
S000"MEIN NAME IST HASE
ein, womit der String mit Bit 7 on gespeichert wird, denn es wird " statt ' verwendet. Betrachten wir zu diesem Zweck erneut nach
S000'MEIN NAME IST HASE
und nach
D
die Hex-Folge, die CD C5 C9 usw. lautet, und dann nach
S000"MEIN NAME IST HASE
und nach
D
die Hex-Folge, die jetzt 4D 45 49 usw. heißt.
Unter ProDOS werden übrigens Dateinamen im Directory sowie Strings in Textfiles stets mit Bit 7 off gespeichert, so daß wir normalerweise für entsprechende Änderungen
S000'...
und nicht
S000"...
verwenden müssen.

10. Wir können natürlich auch Bytes direkt eingeben, und zwar z.B. mit
S000:AA AB AC
was für die Bytes $AA, $AB und $AC steht. Nach
A
sehen wir „*+,", was die Ascii-Werte zu den obigen Hex-Bytes darstellt. Statt
S000:0A 0B 0C
können wir übrigens auch
S000:A B C
eingeben, d.h. eine „0" als „linke Hälfte" eines Bytes kann man unterdrücken. Desgleichen können wir statt
S000:A B C
auch
S0:A B C
eingeben, denn auch bei der S-Option, die den Offset oder Abstand zum ersten Byte des Blockpuffers ausdrückt, können führende Nullen unterdrückt werden.

11. Wir können jedoch noch mehr tun. Mit
S080,0FF,AA
füllen wir den Puffer im Bereich $080 bis $0FF mit dem Byte $AA, d.h. mit lauter Sternchen, wie man nach
A
gut überprüfen kann.
Auf ähnliche Weise läßt sich auch ein Teil des Puffers oder der ganze Puffer mit z.B.
S000,1FF,00
oder kürzer mit
S0,1FF,0
völlig löschen.
Nachdem wir den Puffer gelöscht haben, geben wir
S0:AA
ein, womit ein einzelnes Sternchen zu Beginn des Blockpuffers eingetragen wird.

12. Nun tippen wir
W0117
oder kürzer
W117
oder ganz kurz
W
womit der Inhalt des Blockpuffers in den Block $0117 = 279 der Diskette zurückgeschrieben wird. W ist etwas narrensicherer, denn es schreibt den zuletzt gelesenen Block zurück. Man hätte jedoch auch z.B.
W111
tippen können, womit ein ganz anderer Block, nämlich $0111, überschrieben worden wäre.

13. Wenn Sie einen Drucker in Slot 1 angeschlossen haben, so geben Sie jetzt
PS1
ein, womit sich EDIT Slot 1 merkt. Mit
PD
was für Printer-Dump steht, können wir nunmehr den momentanen Bildschirminhalt ausdrucken. Probieren Sie die beiden möglichen Ausdrucke aus, indem Sie vor
PD
zum einen
A
und zum anderen
D
eingeben.

14. Bislang haben wir auf Blockebene gearbeitet. Nun laden wir mit
§/BENUTZER.DISK
den ersten Block des Volume-Directory von BENUTZER.DISK ein, das nach
A
und
D
entsprechend angezeigt wird. Ferner können wir uns mit
V
die Volume-Bit-Map, also die Blockbelegung der ProDOS-Diskette, ansehen und bei Bedarf über
PD
ausdrucken.
Wir tippen wieder
§/BENUTZER.DISK
und dann
D
für Hex-Dump und ändern jetzt mit
S004:F1
die Länge des Volume-Namens in 1 (dies ist das „1" von „F1") sowie mit
S005'A
den Volume-Namen selbst in „A". Danach schreiben wir mit
W
diesen ersten Volume-Directory-Block zurück. Wir haben damit den Volume-Name der Diskette geändert und können mit
§/V
zwecks Kontrolle das geänderte Volume-Directory einlesen.

15. Wenn wir jetzt
§/A/PRODOS
eingeben, so wird der „erste" Block der Datei PRODOS eingelesen. Genauer gesagt wird nach
D
der Index-Block angezeigt, der etwa so aussieht:
000- 07 09 0A usw.
Mit
R-
gelangen wir auf den physisch ersten Datenblock (= $0007) von PRODOS, und mit
R+
R+
überspringen wird den Indexblock und können den zweiten Datenblock (= $0009) betrachten.

Soweit die Übungsbeispiele. Weitere Feinheiten, insbesondere zur Makrosprache, zum Find-Befehl usw. entnehmen Sie bitte den Teilen 2 und 3 des dreiteiligen EDIT-Beitrags.
us

```
                    PRODOS SECTOR EDITOR V1.Ø

Source: S6,D1     Blockno: $ØØØ2          Target: S6,D1     Blockno: $ØØØ2
-------------------------------------------------------------------------
Ø78- ØØ 27 53 54 41 52 54 55 *** .. .. .. .. .. .. .. ..    .'STARTU
Ø8Ø- 5Ø ØØ ØØ ØØ ØØ ØØ ØØ ØØ *** .. .. .. .. .. .. .. ..    P.......
Ø88- ØØ FC 46 ØØ ØE ØØ 9F 18 *** .. .. 3C .. 18 .. C9 2C    .öF.....
Ø9Ø- ØØ B7 AA ØØ ØØ ØØ ØØ E3 *** .. .. .. .. .. .. .. 21    .7*....c
Ø98- Ø1 Ø8 B7 AA ØØ ØØ Ø2 ØØ *** .. .. 4F A7 .. .. .. ..    ..7*....
ØAØ- 2D 41 44 52 45 53 53 45 *** ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ    -ADRESSE
ØA8- 4E 2E 54 58 54 31 ØØ ØØ *** ØØ ØØ ØØ ØØ ØØ ØØ .. ..    N.TXT1..
ØBØ- Ø4 3D ØØ Ø5 ØØ D1 Ø7 ØØ *** ØØ ØØ .. ØØ .. ØØ ØØ ..    .=...Q..
ØB8- B7 AA ØØ ØØ ØØ ØØ E3 ØØ *** ØØ ØØ .. .. .. .. ØØ ..    7*....c.
ØCØ- ØØ B7 AA ØØ ØØ Ø2 ØØ 2D *** .. ØØ ØØ .. .. ØØ .. ØØ    .7*....-
ØC8- 41 44 52 45 53 53 45 4E *** ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ    ADRESSEN
ØDØ- 2E 54 58 54 32 ØØ ØØ Ø4 *** ØØ ØØ ØØ ØØ ØØ .. .. ØØ    .TXT2...
ØD8- 42 ØØ Ø5 ØØ D1 Ø7 ØØ B7 *** ØØ .. ØØ .. ØØ ØØ .. ØØ    B...Q..7
ØEØ- AA ØØ ØØ ØØ ØØ E3 ØØ ØØ *** ØØ .. .. .. .. ØØ .. ..    *....c..
ØE8- B7 AA ØØ ØØ Ø2 ØØ ØØ 41 *** ØØ ØØ .. .. ØØ .. .. ØØ    7*.....A
ØFØ- 44 44 52 2E 56 45 52 57 *** ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ    DDR.VERW
-------------------------------------------------------------------------
 §/BLANKØØ!H!§/BLANKØ1!C!>>,
```

Abb. 1

```
                    PRODOS SECTOR EDITOR V1.Ø
          Source: S6,D1     Blockno: $ØØØ2
          Target: S6,D1     Blockno: $ØØØ2
-------------------------------------------------------------------
Ø78- ØØ 27 53 54 *** .. .. .. ..    .'ST
Ø7C- 41 52 54 55 *** .. .. .. ..    ARTU
Ø8Ø- 5Ø ØØ ØØ ØØ *** .. .. .. ..    P...
Ø84- ØØ ØØ ØØ ØØ *** .. .. .. ..    ....
Ø88- ØØ FC 46 ØØ *** .. .. 3C ..    .öF.
Ø8C- ØE ØØ 9F 18 *** 18 .. C9 2C    ....
Ø9Ø- ØØ B7 AA ØØ *** .. .. .. ..    .7*.
Ø94- ØØ ØØ ØØ E3 *** .. .. .. 21    ...c
Ø98- Ø1 Ø8 B7 AA *** .. .. 4F A7    ..7*
Ø9C- ØØ ØØ Ø2 ØØ *** .. .. .. ..    ....
ØAØ- 2D 41 44 52 *** ØØ ØØ ØØ ØØ    -ADR
ØA4- 45 53 53 45 *** ØØ ØØ ØØ ØØ    ESSE
ØA8- 4E 2E 54 58 *** ØØ ØØ ØØ ØØ    N.TX
ØAC- 54 31 ØØ ØØ *** ØØ ØØ .. ..    T1..
ØBØ- Ø4 3D ØØ Ø5 *** ØØ ØØ .. ØØ    .=..
ØB4- ØØ D1 Ø7 ØØ *** .. ØØ ØØ ..    .Q..
-------------------------------------------------------------------
 §/BLANKØØ!H!§/BLANKØ1!C>>>>..
```

Abb. 2

peeker Börse

Verkauf Hardware

IIe, 1 Disk, 80Z+64K, Imagewriter, Joyst., Mouse, 30 Bücher, Software... DM 4000,– VB Tel. 0 72 22/4 19 37.

Verkaufe Apple III und Profile, Software, Preis VB Telefon 0 26 41/14 41.

Sharp MZ 80 B. parallel-seriell Schnittstelle, Grafik-Original-floppy, (NP 5.200,–) umfangreiche Software VB DM 2000,–. Tel. 0 26 41/14 41.

AP-20 (IBS 68000, 128K) + Softw. DM 600,–. Bauer, Telefon 0 71 21/4 05 18.

Triumpf-Adler Kugelkopfdrucker 20 Z/Sek., bestes Schriftbild, seriell-parallel, VB DM 500,–

Zu verkaufen: Apple IIe, Monitor neu, Imagewriter I, 80 Zeilen 64 KB Karte, Apple Works VB DM 2900,–. Tel. 08 21/51 00 29.

Verkaufe Apple Zusatzkarten: 16K, Z80 Karten DM 75,– 80 Zeichen Karten DM 180,– 128KB Karten DM 205,–, IIe Komp. Mainboard DM 385,–, 80Z & 64K für IIe DM 75,–, TEAC 54A 1 × 40 T. (Apple) DM 279,–, Komplettsystem IIe DM 1680,–, auch IBM- Karten / Infos kostenlos. Axel Braukmann, Am Sportplatz 51, 4005 Meerbusch-2, T. 0 21 59/36 26

Apple II+, 128K, Z80, 80Z, Centronics 1 LW 40T, 1LW, 2×80T, Monitor, Cherry Tastatur, viel Software, Tel. 0 25 94/55 38, VB DM 2000,–.

Matrix + Typenraddrucker anschlußfertig für alle Computer liefert:
Mail-Shop, Alsterdorfer Str. 201, 2000 Hamburg 60, T. 040/5 11 76 03

256 KB RAM-Card DM 160,– für Apple II+. Durch das Aufstokken Ihrer LC, werden aus 16 KB 256KB. Wird von vielen Progr. unterstützt da Saturn komp. Die RAM-CARD unterstützt weiterhin Ihre Software für die 16 KBLC. Also senden Sie Ihre LC+V-Scheck an A. Müller, Lessingstraße 17, 4000 Düsseldorf.

Siemens PT 80 mit Apple-Interface (Cent.) VB DM 600,–. Telefon 0 26 41/14 41.

Apple IIc, 128K, 2. Laufwerk, Plasmabildschirm, AW, Originalhandbücher (NP DM 6000,–), VB DM 3000,–. Tel. 0 26 41/14 41.

Verkaufe LISA, 1 MB-Ram, 5 MB-Profile, parallel-seriell 7/7 und Bürosystem 2.0 Software, Mac Works sowie ca. 30 Macintosh Programme, Mulitplan, Jazz etc. VB DM 10.000,–. Tel. 0 26 41/14 41.

Gelegenheit: Apple II + 64K komp. UIDEX 80Z-Karte 2×40 Spurlaufw. im Rechnergehäuse, externe Preh Tastatur, Epromer Karte BMC Mon. DM 2000,– FX80 Drucker mit Grafik-Interface DM 1200,– Tel. 0 71 29/50 27.

IIc (12/85), Imagewriter II (12/85), Monitor, Joystick, viele Bücher, Software, Garantie DM 4500,– Balfer **VIA-Karte** (Peeker 5/85), **220V Steuerung,** Software VB DM 250,– Mahr, Tel. 07 11/3 19 42 55.

Apple II e/c komp IBM, Look, 2LW, Drucker, Karte, Modem, Karte RGB, Karte, Eprommer, Monitor, Bücher, Software. Preis VB, Tel. 0 23 84/39 09.

Verkaufe IBS AP17 256K Ram-Karte mit ausführlichem Handbuch. Inc. Software für DOS u. CPM 2.2 Preis auf Anfrage. Tel. 0 52 54/6 88 54.

EPSON FX-80+ mit Grafikinterface Literatur u. Software VB DM 1250,–, Daniel Keng, Tel. 0 21 96/24 63.

Verkaufe APPLE II+ mit Z80, 64KB, Speechcard, 2×40 Track LW, Drucker Wildcard, ca. 40 Disketten + Box, jede Menge Bücher, etc. VB: DM 1995,–, Tel. 06 103/34 900 ab 17h.

Apple IIc, 2.LW, Monitor, Maus, Joystick RefMan, Software (270 Disk) + Lit. 3 Mon. alt, Preis VB Tel. 0 23 61/49 14 30.

Fernschreiberinterface am Gameport m. Programm DM 79,– P. Benner, Hubertusstr. 131, 4150 Krefeld

Apple IIe 128K, Ramdisk 256K, 2 Drives, EPSON FX-80, unheimlich viel Soft- u. Paperware. Tel. 089/80 18 38.

Verschiedene Apple-Erweiterungs-Karten z.B. Ramworks, Accelerator Uhr, Z80, Wandler u.v.a. zu verk. Tel. 02 12/54 93 36 nach 18 Uhr.

Neues 40TRK. SL-Laufwerk mit Apple-Bus, o. Gehäuse, Made in USA, 6 Mon. Garantie, DM 230,–. Tel. 0 64 82/322.

Apple II+, 64K, 3 Laufwerke, Monitor (22 Mhz), div. Karten. Tel. 0 23 51/5 22 57 ab 14.00 Uhr.

2 BASRAM 256 K UCSD DOS CP/M für II/Basis /08 etc. wie neu 1/2 Preis. Tel. 040/2 29 93 91.

Apple IIc, Maus, Philips-Monitor + Softw. + umfangreiche Literatur für DM 1900,–. Tel. 02 11/22 33 90

Macintosh 128 auf 512KB DM 299,–. Fa. Schlösser, Tel. ab 17.00 Uhr, 089 - 98 58 89

Ankauf Hardware

Abrechnungssoftware (Krankenkassen) **von der KBV zugelassen,** Lightpenbedienung gesucht. Zahle bis DM 4000,–. Tel. 0 26 41/14 41.

Suche Apple III-System. Tel. 0 26 41/14 41.

Verkauf Software

IIe+c: CAD paint: exakte Zeichnungen m. Beschriftung (2 Stärken), 40 Funktionen, 80 fertige Symbole für Wohnungspläne + elektr. Schalt. Info 2 DM, Briefm., Disk: DM 99,– Scheck. Lohmann, M.-Müller-Ring 7, 6500 Mainz

SUPERPROGRAMME FÜR APPLE-2 (R) Lottoprogramm LOT 2000 CP/M DM 29,90 Monitorprg. (MEMDIT-1) (DOS) DM 24,90, Prg. für dBASE II 2,4 (Quelltext), Dateibearbeitung (DB-DATEI) DM 10,– Programmierung (DB-PROGRAMM DM 10,– DB-Prg. werden als Listing geliefert sonst Apple-CP/M-Disk u.P/V incl. M. Filbrich (Abt. Softcontrol), Hauptstr. 22, 3171 Adenbüttel Tel. 0 53 36/82 59.

Software Uhr für Apple II+, e, c, Zeitschaltmöglichkeit Diskette + Anleitung DM 25,– Oecking Tel: Do. 02 31 / 39 19 20

Utilities-Tools-Grafik: prof. Prg. für Apple II+, Info gegen Rückporto bei: Büttner, Brüder-Grimm-Str. 4, 3180 Wolfsburg 12.

Sonderangebot
Disketten 3 1/2", 1DD 135 TPI, DM 6,00, Disketten 3 1/2", 2DD 135 TPI DM 7,50, Disketten 5 1/4", 1D 48 TPI, DM 2,10, Disketten 5 1/4", 2D 48 TPI, DM 2,30, PL-Box m. Schloss f. 100 St. DM 28,–. Alle Disketten m. Garantie und Verstärkungsring. Fa. Allg. Austro Agentur, Ringstr. 10, D-8057 Eching, Telefon 0 81 33/61 16, Tlx. 5 27 551.

!!!! Die Supergelegenheit !!!! Top-Apple-Games (engl.): Ultima II 29,–, Wizardry I 29,–, Wizardry II 25,–, The Heist 9,–, Wizard & Prince 8,–, Ulysses 19,–, Time Zone 39,–, Dino Eggs 19,–, Miner 2049er 19,–, Wiziprint 9,–, Poli. Art. 19,–, Crypt of Medea 29,–, Starmaze 9,–, Galactic Attack 9,–, Mission Asteroid 19,–.
Schnell bestellen: 089/93 98 94! Nur so lange die Vorräte reichen FUN*TASTIC-MailOrder GmbH.

MEGABOARD Version A (CP/M 3.0): jetzt endlich mit Pro-DOS-Anpassung durch neuentwickelte Patch-SW Änderung gegen Einsendung der alten SW und DM 50,– an Matthias Meyer, Postf. 191, CH-9001 St. Gallen.

APPLE II: DFÜ-Kermit, Pascal satt, Public Domain in DOS v.CP/M. Je Volume DM 15,–. Bahnhofssimulation, Sprachen (S.A.L.), Schulprogr. Gratisinfo: Fa. Waltraud Muhle, Waldwinkel 3, 2105 Seevetal 3.

65C02-TRACER + Debugger mit windowtechnik, 176S. Handbuch Anfänger: Einstieg in ASSEMBLER Fortg.: Leistungsfähiger DEBUGGER DM 79,–. Telefon 0 70 71/7 45 44, Stimmler Elektr., 7400 Tübingen, Heinlenstr. 34.

Apple-II. Frei-Prg.-Liste anf. b. Fa. Josef Gebhardt, PF 1174, 8458 Sulzbach-Rosenbg. Tel. 0 96 61/5 29 16.

Das neue Mittwochslotto zu DM 25,– Der Turm von Hanoi zu DM 10,– Förter, Am Wetterbach 53, 75 Karlsruhe 41.

Apple II. DFÜ-Kermit, Pascal satt, Lisp, Public Domain in DOS und CP/M: Je Volume DM 15,– Schulpro. Gratisinfo: Fa. Waltraud Muhle, Waldwinkel3, 2105 Seevetal3

Für Ihre Unterlagen

Abonnement bestellt

am: _____

Vertrauensgarantie:
Ich habe davon Kenntnis genommen, daß ich die Bestellung schriftlich durch Mitteilung an den Dr. Alfred Hüthig Verlag GmbH, Postfach 10 28 69, 6900 Heidelberg innerhalb von 7 Tagen widerrufen kann. Zur Fristwahrung genügt die rechtzeitige Absendung des Widerrufs (Datum des Poststempels).

Peeker
Leserservice
Postfach 10 28 69
6900 Heidelberg

Für Ihre Unterlagen

Folgende Bücher bestellt:

_____

_____

_____

_____

_____

am: _____

bei:

Peeker
Versandbuchhandlung
Postfach 10 28 69
6900 Heidelberg 1

Für Ihre Unterlagen

Folgende Disketten
und Programme bestellt:

_____

_____

_____

_____

_____

am: _____

bei:

Peeker
Softwareabteilung
Postfach 10 28 69
6900 Heidelberg 1

peeker

Abo-Karte

Ja, ich möchte **Peeker** abonnieren.
Liefern Sie mir **Peeker** ab Ausgabe ..... zum Jahresbezugspreis von z. Zt. DM 72,– (Inland) inkl. MwSt. Die Lieferung erfolgt frei Haus. Porto, Verpackung und Zustellgebühren übernimmt der Verlag. Der Jahresbezugspreis für das Ausland beträgt z. Zt. DM 72,– plus DM 18,– Versandspesen.

X

Datum 1. Unterschrift

Bitte lesen!

Vertrauensgarantie: Ich habe davon Kenntnis genommen, daß ich die Bestellung schriftlich durch Mitteilung an den Dr. Alfred Hüthig Verlag GmbH, Postfach 10 28 69, 6900 Heidelberg innerhalb von 7 Tagen widerrufen kann. Zur Fristwahrung genügt die rechtzeitige Absendung des Widerrufs (Datum des Poststempels).

X

Datum 2. Unterschrift

Verlagshinweis: Das Abonnement verlängert sich zu den jeweils gültigen Bedingungen um ein Jahr, wenn es nicht 2 Monate vor Jahresende schriftlich gekündigt wird.

Wir können nur Bestellungen mit zwei Unterschriften bearbeiten.

peeker

Buch-Karte

Bitte senden Sie mir gegen Rechnung folgende Bücher:

- ☐ Bühler, Applesoft-BASIC, 3-7785-1094-0, DM 38,–
- ☐ Eggerich, dBase II, Bd. 1, 3-7785-1147-5, DM 39,80
- ☐ Eggerich, dBase II, Bd. 2, 3-7785-0987-X, DM 39,80
- ☐ Eggerich, dBase II, Bd. 3, 3-7785-0988-8, ca. DM 40,–
- ☐ Gabriel, Applewriter, 3-7785-1234-X, DM 35,–
- ☐ Hagemüller, Microsoft-BASIC, Bd. 1, 3-7785-1038-X
- ☐ Juhnke/Redlin, Apple Pascal, Bd. 1, 3-7785-1246-3, ca. DM 40,–
- ☐ Kehrel, Apple Assembler lernen, Bd. 1, 3-7785-1151-3, DM 38,–
- ☐ Kehrel, Apple Assembler lernen, Bd. 2, 3-7785-1170-X, ca. DM 38,–
- ☐ Schäpers, ProDOS Analyse, 3-7785-1134-3, DM 68,–
- ☐ Schäpers, Bewegte Apple-Graphik, 3-7785-1150-5, DM 58,–
- ☐ Stiehl, Apple DOS 3.3, 3-7785-1297-8, DM 28,–
- ☐ Stiehl, Apple ProDOS, Bd. 1, 3-7785-1098-3, DM 28,–
- ☐ Stiehl, Apple ProDOS, Bd. 2, 3-7785-1036-3, DM 30,–
- ☐ Stiehl, Apple Assembler, 3-7785-1047-9, DM 34,–
- ☐ Wassermann, Apple IIc Handbuch, 3-7785-1157-2, DM 35,–

Datum Unterschrift

peeker

Software-Karte

Bitte senden Sie mir gegen Rechnung folgende Disketten:

- ☐ Peeker-Sammeldiskette, einzeln
  Disk# _____, Disk# _____
  Disk# _____, Disk# _____
  Peeker je Disk DM 28,– (einzeln)
- ☐ Peeker-Sammeldiskette, im Fortsetzungsbezug ab Disk# _____
  (Mindestbezug 6 Disketten)
  Preis je Disk DM 20,–
- ☐ Apple DOS 3.3, Begleitdisk., DM 28,–
- ☐ ProDOS, Band 1, Begleitdisk., DM 28,–
- ☐ ProDOS, Band 2, Begleitdisk., DM 28,–
- ☐ Apple Assembler, Begleitdisk., DM 28,–
- ☐ ProDOS-Editor 1.0, Programm, DM 98,–
- ☐ MMU 2.0, Programm, DM 98,–
- ☐ INPUT 2.0, Programm, DM 98,–
- ☐ Softbreaker 1.0, Programm, DM 48,–
- ☐ DB-Meister, Programm, DM 290,–
- ☐ Superplot, Programm, DM 48,–
- ☐ Superquick, Programm, DM 48,–
- ☐ Turtle Graphics, Programm, DM 98,–
- ☐ Disk 40, Programm, DM 48,–
- ☐ Kyan-Pascal 2.0, Programm, DM 170,–
- ☐ Fast-Writer, Programm, DM 98,–

Datum Unterschrift

Abo-Karte

Name

Firma

Straße

PLZ/Ort

Ich wünsche jährliche Berechnung durch:
☐ Verlagsrechnung ☐ Abbuchung von meinem Bank- bzw. Postscheckkonto

Bank/PschA

Bankleitzahl

Kto.-Nr.

Buch-Karte

Karte bitte vollständig ausfüllen

Vorname, Name

Firma

Straße

PLZ/Ort

Telefon mit Vorwahl

Software-Karte

Karte bitte vollständig ausfüllen

Vorname, Name

Firma

Straße

PLZ/Ort

Telefon mit Vorwahl

POSTKARTE

Peeker
Leserservice

Dr. Alfred Hüthig Verlag GmbH

Postfach 10 28 69

6900 Heidelberg

POSTKARTE

Peeker
Buchabteilung

Dr. Alfred Hüthig Verlag

Postfach 10 28 69

6900 Heidelberg 1

POSTKARTE

Peeker
Softwareabteilung

Dr. Alfred Hüthig Verlag

Postfach 10 28 69

6900 Heidelberg 1

INPUT 2.0

Ein Bildschirm-Maskengenerator für DOS 3.3 und ProDOS

von U. Stiehl

1984, Diskette und Manual, DM 98,–
ISBN 3-7785-1021-5

„Input 2.0" liegt wahlweise in der Bank 1 oder Bank 2 der Language Card und wird durch einen kurzen Driver in den unteren 48K aufgerufen.
Für jedes Feld der Bildschirmmaske lassen sich u. a. definieren: Feldlänge (bis zu 255 Zeichen) – Vtab – Htab – Datentyp (insgesamt 8 Typen) – Scrollflag (starre oder dynamische Maske) – Ctrlflag – Füllflag – Löschflag – Bildschirmflag (40- oder 80-Z-Darstellung). Innerhalb eines Eingabefeldes besteht jeder denkbare Redigierkomfort (Insert, Delete, Rubout, Restore usw.).
Gerätevoraussetzung: Apple IIe oder IIc; ferner Apple II+ im 40-Zeichenmodus

MMU 2.0

Memory Managements Utilities

für die Apple IIe 64K-Karte DOS 3.3 (und ProDOS)

von U. Stiehl

1984, Diskette und Manual, DM 98,–
ISBN 3-7787-1023-1

Insgesamt enthält die neue „MMU 2.0"-Diskette über 25 Programme, die neue Einsatzmöglichkeiten für die Extended 80 Column Card (erweiterte 80-Z-Karte = 64K-Karte für den Apple IIe) erschließen. Ein Teil der Programme laufen auch auf dem Apple II Plus, doch ist „MMU 2.0" primär für 64K-Karte-Besitzer gedacht.

Gerätevoraussetzung: Apple IIe mit 64K-Karte oder IIc

Softbreaker 1.0

Eine softwaremäßige Interrupt-Utility für die Apple IIe 64K-Karte

von U. Stiehl

1984, Diskette und Manual, DM 48,–
ISBN 3-7785-1022-3

Softbreaker ist ein Assemblerprogramm, mit dessen Hilfe Programme, die sich von der 64K-Karte (= Extended 80 Column Card für den Apple IIe) starten lassen, unterbrochen, gespeichert, geladen und exakt an der Stelle der Unterbrechung fortgeführt werden können. Dadurch ist es auch möglich, Sicherungskopien von sogenannten kopiergeschützten Programmen herzustellen.
Mit Softbreaker unterbrochene Programme werden komplett, d. h. die ganzen 64K einschließlich Language Card, in nur ca. 11 Sekunden auf einer formatierten Diskette gesichert.
Gerätevoraussetzung: Apple IIe mit 64K-Karte, nicht IIc, nicht neue ROMs

Hüthig Software Service, Postfach 10 28 69, D-6900 Heidelberg

PEEKER Börse

Verkauf Software

Apple II: Verkaufe Topsoftware! z.B.: Ballblazer DM 110,– Merlin pro Z50,– Kundenverwaltung 50,– Info (0,80) bei: W. Rittmeyer, Wehrbruchweg 30, 4060 Viersen 1.

Pascal, KYAN u. Turbo 3.0, org., Manuals, neu DM 300,–. Telefon 0 26 66/704.

Profi-Software für den Apple II/+ /e/c. Fakturierung DM 380,–, Kundenverwaltung DM 210,–, Lagerverwaltung DM 380,–, alle Programme sind verknüpfbar, Liste gegen DM 2,– in BRM. Demodisk DM 50,–. Tel. 02 11/7 33 77 78.

Endlich lieferbar! BABEL II der interaktive Debugger mit vielen Hilfsfunktionen und ausführl. deutsch. Handbuch. Lieferung per Rechnung ab Mai 1986, DM 37,90, Ruf, Clemensstr. 74, 8 München 40

AWorks-Patch f. Epson-Interface DM 20,–; Steffen Holzinger, Berliner Str. 62, 6301 Biebertal.

Vokabeltrainer 1.0 Apple II Preisträger Jugend forscht '85 Karteik. sys., Lexikonfunktion Info DM 2,– B. Beckert, Zähringerallee 37, 7530 Pforzheim 1.

Window-Technik für APPLESOFT bei allen Apple II-Typen DM 35,–, Heinz, Waldgürtel 7, 5060 Berg. Gladbach 1

Jane u. Dazz/e Draw (Orig.) zu verk. DM 250,–. Telefon 0 28 37/84 53 ab 17 Uhr.

Ankauf Software

Suche Spiele für Apple IIe, Tel. 0 22 04/6 82 31.

Suche Init / Treiber usw. Für BASIS 256K Ramkarte/IIe/UCSD PASCAL 1.2 (evtl. auch DOS 3.3). Ingram, Lütticherstr. 30, Köln 1.

CP/M 2.2 Systemdiskette gesucht für IIe A. Grün Telefon 0 21 51/30 21 77

Handwerkerprogramm/ Schreiner auf IIe. W. Müller, Sandhäuser Str. 1, 6900 Heidelberg, Tel. 0 62 21/78 11 32.

Verschiedenes

APPLE REPARATUREN
(auch compatible M-boards, z.B. **Atlas, Arca, CES, Datastar, Dipa, Lasar, Mewa, PC-48 + 64, Plato, Radix, o. ae.) sowie Zusatzkarten und Disk-Drives** führt unser Spezialistenteam mit mehr als 5-jähriger Kunden- und Reparatur-Dienst-Erfahrung, garantiert zuverlässig und besonders kostengünstig aus.
Bitte genaue Fehlerangabe sowie Tel. Nr. für evtl. Rückfragen nicht vergessen.
Auf Wunsch Kostenvoranschlag.
aaa-electronic gmbh
Habsburgerstr. 134, 7800 Freiburg, Tel. 0761/276864, Tx. 772642aaad

DFÜ-Szene CH: die interessantesten Auszüge aus schweizer Mailboxen mit vielen Tips und Tricks auf randvoller Apple-Disk gegen Einsendung von DM/Fr. 10,– an Matthias Meyer, Postf. 191, CH-9001 St. Gallen.

Apple IIc + Brother M-1009 wer kennt Konfiguration für Text + Grafik (seriell)? Norbert M. Dörr, H.-Heine-Str. 1. 6307 Linden. Tel. 0 64 03/84 79.

!!!! Da gibt's was umsonst !!!! 4 × im Jahr den neuen Katalog. Bühler Elektronik, Postfach 32, 7570 Baden-Baden

Apple III Bes. wg. Erfahrungs/Prg. Austausch usw. bt. melden, Eckl, Bichlerstr. 4b, 8122 Penzberg.

Für weitere Informationen zu einem der in dieser Ausgabe vorgestellten Produkte stehen Ihnen die Produktkarten zur Verfügung.

Bitte verwenden Sie für Kleinanzeigen die vorgedruckten Antwortkarten in diesem Heft.

Apple und IBM kompatible Computer

16K, Z80, Diskcontroler je 98,–
80 Zeichenkarte mit Softswitch
2 Zeichensätze 198,–
IIe-kompatibles Motherboard
ohne Firmware 498,–
Erphi-controler mit Autopatch 285,–
TEAC FD-54A mit Apple-Bus 298,–
TEAC FD-55B-V 2 × 40 Track 395,–
TEAC FD-55F-V 2 × 80 Track 448,–
FD4 Spezialcontroler für Laufwerke
mit bis zu 2 × 80 Track 120,–
OLYMPIA compact NP 1298,–
Monochrome Monitore . . . . . ab 375,–
Farbmonitore . . . . . ab 998,–
Tastaturen für IBM und Apple . . . ab 298,–
512K-RAM-Karte mit 256 K bestückt 298,–
Apple Super-Modem-Karte inkl. dt. Software und dt. Handbuch 348,–
Versand nur per Nachnahme oder Vorkasse.
Weiteres Zubehör für Apple und IBM gegen frankierten Rückumschlag.
128K Karte (Saturn kompatibel) . **248,–**

Ulf Mohwinkel Electronic
Berliner Straße 73 Pf: 250 166
5090 Leverkusen Fettehenne
Telefon 02 14/9 37 81 od. 9 50 60

erphi rφ erphi electronic

640KB Controller und Floppy für Apple II+/e und kompatible.

Generalvertretung für die Schweiz und **FL**

(Händleranfragen erwünscht)

beltronic

Im Chapf CH-8455 Rüdlingen
Tel: 01-8 67 31 41 · Tlx 8 25 981

Weiter führen wir:
IBM kompatible PC ab 2400,–
Apple IIe kompatible ab 1200,–

aaa electronic gmbh

Neue Preise # 6

IMB-PC/XT-COMPATIBLE COMPUTER + ZUBEHÖR
Mainboard XT I. 16bit + 256K-RAM + 8 Slots, Leerkarte 79,–
Mainboard XT-640 K (on Bord) f. 16 Bit + 8 Stots Leerkarte 89,–
Disk Controller leer (f. 4 x 360KB-Drives) 34,–
Monochrome Graph. + par. Printer Leerkarte (Hercules comp.) 55,–
Color RGB + Video Graphik Leerkarte 55,–
Parallel Printer Leerkarte 29,–
512K-RAM Leerkarte 55,–
Disk-Muliti I/O Leerkarte, (f. 3x360 K-Drive, 2xRS 232 1xpar. Printer Int., Game/Joystick Port, Real Time Clock/Kalender gepuff.) 59,–
384 K-Multifunktions Leerkarte, (RS 232, par. Printer Int., Game/Joystick Port, Real/time Clock/Kalender gepuffert) 55,–
Prototype/Lochraster Leerkarte, durchkontakiert 69,–
Mainboard XT-256 f. 16 bit CPU, OK (bis 256K-on board, oder 640 K mit Addr. auf Karte aufr.) + Boot-ROM + 8 Slots best. + geprüft 315,–
Mainboard XT-640 f. 16 bit CPU, OK (bis 640 K-on board aufr.) 327,–
Mainboard XT-640 w.o. jedoch 256 K-RAM best. + gepr. 417,–
Mainborard XT-640 w.o. jedoch 640 K – vollbest. und geprüft 587,–
64KB-RAM-Chip-Aufrüstsatz (9 Stück) einzeln geprüft 42,–
256 KB-RAM-Chip-Aufrüstung (9 Stück) einzeln geprüft 97,–
Disk Controller gepr. (f. 4 x 360 KB-Drives)(Extra Kabel = 39,–) 109,–
Monochrome Karte geprüft 235,–
Monochrome Graph. + par. Printer Karte gepr. (Hercules comp.) 268,–
Color RGB + Video Graphikkarte gepr. 166,–
Parallel Printer-Karte gepr. o. Kabel (Extra Kabel = 49,–) 77,–
512K-RAM-Karte gepr. (0KB) 129,–
384K-Multifunktionskarte, RS 232, par. Printer, Interf. Game/Joystick Port, Real Time Clock gepuff., 0K + RAM-Disk/Spooler/Uhr Softw. 325,–
Disk Multi I/O Karte (f. 2x360 K-Drive, 2xRS 232, 1xpar. Printer Int., Game/Joystick Port, Real Time Clock/Kalender gepuff.) 388,–
Profi-Funktionstastatur ASCII + 15er Block ab 215,–/(dt. ab 245,–
135W/150W-Netzteil mit eingebautem Ventilator, Kurzschlußf. 219,–/279,–
360KB-Diskdrive Slimline (Track Access Time <2 m S), R/W gepr. 319,–
Joystick f. IBM + Compatible 79,–
Maus incl. Software f. IBM + Compatible 299,–
PC/XT – KOMPLETT-SYSTEME EINGEBAUT UND GEPRÜFT
XT-640, 16 bit-Rechner (incl. Boot-Eprom) 256K-Ram im IBM-look-like Met.-Gehäuse + Profi-Fkt. Tast. + 135 W-Netzt. gepr. 999,–
XT-640, w.o. + Contr. +360KBDrive + Color RGB + Vid. Graphic-Karte 1.677,–
XT-640, w.o. jedoch statt Color mit Monochrome Karte 1.737,–
XT-640, w.o. jedoch mit Monochrome-Graphic-Printer-Karte 1.817,–
Aufpreis für XT-640 M-board mit 640 K vollbestückt + gepr. 169,–
20 MB-NEC-Festplatte + DTC-Controller mit Kabel + Man., geprüft 2375,–

PC 10-Commodore 512 K+2 Laufwerke + Monitor + dt. Tast. 3.876,–
PC 10-512 K + 1 Laufwerk + 20 MB-Festplatte 5969,–
640 K-Speichererweiterungs-Set für PC 10 279,–
PHOENIX GENIE 16 C -PC-compatibel

APPLE-BUS COMPUTERPLATINEN + PERIPHERIE
Motherboard II 48K leer, durchkont. mit Lotstopplack 57,–
Motherboard II 64 K + Z 80 CPU, leer 95,–
M-board II 48K, gesockelt, vollbest. + gepr. 337,–
Motherboard II 64K gesockelt, vollbest. + gepr. 375,–
M-board II 64K + Z80 CPU, gesockelt, v. best. + gepr. 378,–
M-board IIe 64K vollbestückt + gepr. 435,–
APOLLO II **Kunststoff-Leergehäuse** 85,–
APOLLO II **Gehäuse** incl. Funkt.-Tast. 217,–
Superstarkes Netzteil – Kurzschlußfest 5A=115,–/7A= 145,–
Leerplatine: 16K-RAM, Z80/CPU-, Controller DOS 3.3, Parallel-Druckerkarte, Ser. Int. V24, 80Z/24Z-Karte, PAL-Karte je 18,95
Entwicklungslochraster-Leerplatine durchkontaktiert 27,–
16K-Speicher-Karte geprüft 55,–
APPLE orig. 16K-Language Karte gepr. 98,–
Z-80 CPU-Karte geprüft 64,–
Controller DOS 3.3-Karte geprüft 63,–
Auto-Controller DOS 3.2/3.3-Karte gepr. 97,–
Erphi AFDC 2 Controller + Autopatch Softw. + Handbuch 278,–
Parallel-Drucker-Karte geprüft 74,–
Par.-Drucker-Grappler comp. Leerplatine 27,–
Par.-Drucker Karte-Grappler comp. geprüft (Extra Kabel=45,–) 169,–
8bit par. Int. + 64K-Buffer + Kabel gepr. (Grappler comp.) 299,–
Seriell-Interface-Karte V24 geprüft 88,–
Super-Seriell Leerkarte 34,–
Super-Seriell-Interface Karte gepr. 169,–
80-Zeichen/24-Zeilen-Karte geprüft 119,–
80Z/24Z-Karte + Softswitch-Schalter Leerplatine 24,–
80Z/24Z-Karte + Softswitch-Schalter, gest. Scharf geprüft 127,–
80Z/24Z-Softwitch Karte gepr. für IIe 49,–
80Z/24Z + 64K-Ram + Softsw. Leerkarte f. IIe 27,–
80Z/24Z + 64K-Ram + Softsw.-Karte best. + gepr. f. IIe 87,–
IEEE-488 Int. Karte gepr. (Extra Kabel=45,–) 297,–
PAL-Modulator/Color-Karte geprüft 99,–
RGB-Color Karte gepr. 129,–
128K-Speicher-Leerkarte 29,–
128K-Speicher-Leerkarte + Software + Manual 99,–
128K-Speicher-Karte + Software geprüft + Manual 264,–
Maus incl. Software 179,–
EPROM-Burner (2716/32/64) geprüft 109,–
6522-VIA Karte geprüft 135,–
Clock Karte geprüft 94,–
Speech Karte geprüft 53,–
Wild Karte geprüft 59,–
AD/DA-8Bit Karte gepr. 375,–
UHF/TV-Modulator universell 27,50
Lüfter anclipsbar (220 V) 79,–
Dataphon s21d Akustikkoppler-FTZ mit RS 232/V24 Anschl. 275,–

APOLLO//(48K) + UHF-Mod. + Gr/Kl vollbest. + gepr. 617,–
APOLLO//(48K) + Disk II F + Contr. + 12"-Monitor 1.285,–
APOLLO//A (48K) + Gr/Kl + 15er-Tastatur vollbest. + gepr. 677,–
APOLLO//A (48K) + Disk II F + Contr. + 12"-Mon. 1.345,–
APOLLO// ASKF (48K) vollbest. + gepr. Sep. Keyboard + Gr/Kl + 15er-Block mit Funkt.-Tasten im „IBM-look like" Gehäuse 835,–
APOLLO// ASKF (48K) w.o. + Disk II F + Contr. 1.179,–
Apollo II ASFK (48K) w.o. + 2xDisk II-F+Contr. + 12" Mon. 1.787,–
Aufpreis f.o. Systeme von 48K zu 64K 55,–
Aufpreis von 48K zu 64K+280 CPU 89,–
Apollo IIe (64K) vollbest. und geprüft 725,–
Apollo IIe (128K) + 80Z/24Z-Karte geprüft 815,–
Apple IIe (64K) + 15er-Tastatur vollbestückt 819,–
Apollo IIe ASKF (64 K) vollbestückt und geprüft ab 876,–
Apollo IIe (64K) + Disk II F+12" Monitor 1.409,–

APPLE IIe 128K + 80Z/24Z Karte + Softw. 1.998,–
Disk-Drives voll APPLE II, 100%ig compatibel + R/W geprüft
Disk//+Contr. + Kabel DOS 3.3 (Siemens) im Geh. 529,–
Disk//(Siemens) im Geh. + Kabel 467,–
Disk//F (1/2 Höhe) im Gehäuse + Contr. + Kabel 349,–
Disk//F (1/2 Höhe) im Gehäuse 288,–
Disk//F-HS (High Speed 1/2 Höhe) im Gehäuse + Contr. + Kabel 379,–
Disk//F-HS (High Speed 1/2 Höhe) im Geh. +Kabel (Track Ac.<2mS) 318,–
Disk//F-HS-80 Track im Gehäuse + Kabel f. orig. Contr. geeignet 389,–
Disk//F-HS-80 w.o. + Contr. + DOS 3.3 + CP/M Patch-Softw. 489,–
Erphi DuoDisk 1,2 MByte im Gehäuse + Erphi-AFDC 3-Contr. 1.195,–

APPLESOFT + Turorial + Reference Manual engl. 75,–
APPLE-DOS 3.3 Manual engl. 25,–
APPLE-DOS 3.3 Hb. dt. 45,–
APPLE-Pascal Reference + Operating Manual engl. 48,–
APPLE-Pascal Reference Hb. dt. 49,–
APPLE-Pascal Language Hb. dt. 39,–
APPLE II-Pascal Hb. dt. (te-wi) 59,–
APPLE-Fortran Manual engl. 69,–
CP/M-Softcard Vol. I + II, Manual engl. 69,–
APPLE II + IIe-Anwenderhandbuch deutsch (te-wi) 66,–

Z80 Karte gepr. f. APPLE/2c + RAM-Disk-Software + Hb. 489,–
Macintosh Umrüstung von 128K auf 512K 567,–

Epson LX 80, 8 Bit/par. (Extra Traktor = 69,–) 767,–
EPSON FX 85F/T+, 8bit/parallel 160 Z/S. NLQ + IBM Komp. 1.275,–
EPSON FX 105+, 8bit/parallel 15" sonst w.o. 1.696,–
EPSON Traktoraufsatz für FX 80 + FX 85 129,50
APPLE/EPSON-Drucker Graphic-Interface + Kabel 169,–
MX 80/82, FX/RX 80 Spezialfarbband-Kassette 15,95

BROTHER-Typenraddr. HR 15XL 8bit/par. neuestes Mod. 1.178,–

Wir führen verschiedene Monitore von Zenith, Phillips und Sanyo mit Video – TTL (IBM Komp.) oder Coloreingang. – Bitte Preisliste anfordern! –

Disketten in Box + Aufkleber I. Wahl, 10er Pack/100er Pack./ Stückpreis:

5¼" Verstärkt	BASF	3M SCOTCH	FUJI	DATA MAGNETICS	NEUTRAL
1X, SS/SD	3,49/ 3,29				
1D, SS/DD	3,69/ 3,49	3,38/ 2,98	4,27/ 3,97	2,56/2,46	2,25/2,15
2D, DS/DD	4,89/ 4,59	4,58/ 3,98	5,87/ 5,47	3,26/3,06	2,75/2,55
1D/96TPI	4,59/ 4,39	4,88/ 4,68			
2D/96TPI	5,39/ 5,09	5,48/ 5,18	7,87/ 7,67		
2HD/1.2MB	12,49/11,59	8,98/ 8,18	12,87/12,57		
1X, Hard	4,29/ 4,19				
8"					
1X, SS/SD	4,49/ 4,39	6,68/ 6,48	6,97/ 6,67		
1D, SS/DD	4,79/ 4,59	8,58/ 8,28			
2D, DS/DD	7,09/ 6,89	9,88/ 9,58	10,17/ 9,77		
1X, Hard	**4,49/ 4,39**	**6,68/ 6,48**			
3½"					
Einseitig	7,29/ 6,89	6,78/ 6,08	7,67/ 7,17		
Zweiseitig	11,59/11,19	9,28/ 8,48	11,17/ 9,97		

Das sind Preise!

3M-Scotch 5¼" Flüssig-Reinigungsset (2 Disk.) zweiseitig 48,95
5¼" Disketten-Archivbox für 10 Disks (1 St./10 St.) 4,75/43,50
5¼" Disk-Karteikasten Kunststoff (ca. 80 Disk) 26,50
5¼" Disk-Karteikasten Kunststoff (ca. 80 Disk) Rauchglas 39,95
5¼" Disk-K.K.Rauchglas, abschließb. (f. ca. 80-100 Disks) 47,50
2000 Bl. Tabpapier (24 cm x 12" einf.) weiß oder grün/weiß perf. 45,–
4000 Et.-Aufkl. doppelr. (107 x 36 mm auf 240 x 12" perf. Trägerp.) 66,50
4000 Etik.-Aufkl. einreih. (107 x 36 mm auf 125 x 12" perf. Träg.) 65,–

OSZILLOSCOPE HAMEG ab Lager

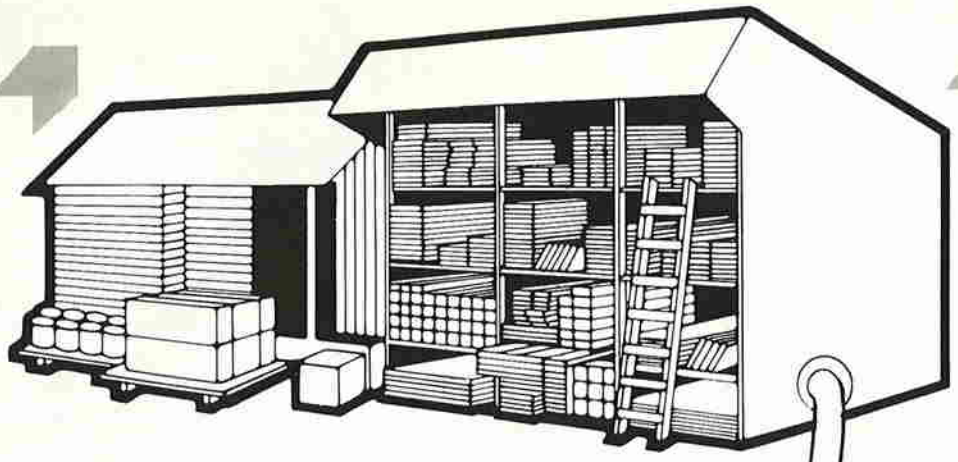
Bei Vorauszahlung frei Empfangsstation unversichert in der BRD, ausgenommen Papier und Etiketten, sonst N.N. + V.S. ab DM 30,–– Preise incl. MwSt.

Öffnungszeiten: Mo, Di, Do, Fr v. 10-18 h, Mi u. Sa v. 10-14 h, Ia Sa v. 10-18 h.
Telef. Best.: Mo, Di, Do, Fr von 10-19 h, Mi u. Sa wie Öffnungszeiten.

GEWÄHRLEISTUNG: 6 Monate, auf alle bei uns gekauften Geräte, durch unsere eigene Service-Werkstatt.

REPARATUREN an Apple + Compatible Geräte + Zubehör führt unser Spezialistenteam garantiert zuverlässig + besonders kostengünstig aus.
Sprechen sie mit uns. Kostenvorschlag auf Wunsch!

aaa electronic gmbh
Telex: 07 72 642 aaa-d
Habsburgerstraße 134
7800 FREIBURG, Tel. (07 61) 27 68 64
Bauelemente – Bausatze – µP's
Meßgerate – Zubehor – Fachliteratur
Fachgeschäft fur Elektronik + Mikrocomputer



Megaboard- und AFDC-Massenspeicher

Mit ProDOS-Backup-Programmen für die MDB-Festplatte und das Erphi-Subsystem

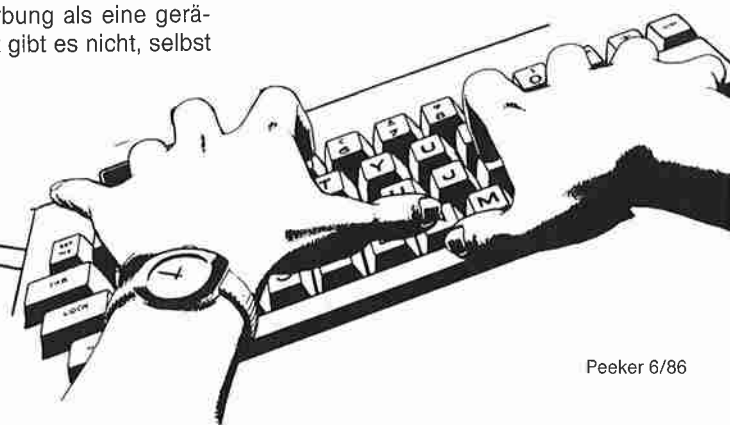
von Ulrich Stiehl

Hinweis: Die weiter unten gelisteten Back-up-Programme können in modifizierter Form auch bei anderen Fabrikaten als Floppy-Harddisk-Kopierprogramme sowie darüber hinaus als Floppy-Floppy-Kopierprogramme verwendet werden.

1. Disk-II-Nachfolger

Die Besitzer von Disk-II- oder Duodisk-Laufwerken werden sich langfristig nach neuen Laufwerken umsehen müssen, da die alten 35-Spur-Drives seit Anfang 1986 auslaufen. Dies ist insofern bedauerlich, als bislang alle kommerziellen Apple-II-Programme auf 35-Spur-Disketten ausgeliefert wurden. Es wird allerdings nach wie vor funktionsgleiche Laufwerke von Fremdherstellern geben, die zudem meist als 40-Spur-Drives ausgelegt sind. Von Apple selbst sind in Zukunft nur noch 3,5-Zoll-Drives erhältlich, von denen wir ein Testexemplar auf der „Systems"-Messe im Herbst 1985 angefordert haben. Jetzt im April 1986 wurde das Drive genehmigt, da das Werbe-Kontingent, wie es hieß, „vorübergehend" erschöpft sei. Und da vor kurzem der Geschäftsführer R.M.Deja sowie der Controller (= Finanzchef) E. Schmollinger bei Apple „freigestellt" wurden, könnte es mit dem Test-Drive noch länger dauern. All dies ist lächerlich, kleinkariert und kurzsichtig. Beispielsweise haben wir von Atari den 520 St bereits zu einem Zeitpunkt erhalten, als das Gerät in Deutschland noch gar nicht auf dem Markt war. Wenn Apple-München nicht einmal der größten deutschen Apple-Zeitschrift Testgeräte überlassen kann (aus finanziellen Gründen) oder überlassen will (aus politischen Gründen), dann liegt das Marketing bereits derart im argen, daß man sich nicht mehr zu wundern braucht, wenn jedes Jahr die Köpfe rollen, denn eine billigere indirekte Werbung als eine gerätespezifische Fachzeit gibt es nicht, selbst wenn im Peeker gelegentlich Kritik geübt wird. Die „Infowelt", Heft 14/1986, drückte es in den Beiträgen „Image und Umsatz futsch" und „Apple auf Schleuderkurs" erheblich drastischer aus: Danach soll, was von Apple nicht dementiert wurde, der BRD-Umsatz von 92 Mio. im Geschäftsjahr weit hinter dem Plansoll von 140 Mio. zurückgeblieben sein, und man peile für 1986 nur noch 65 Mio. an.

Wir werden uns deshalb im Peeker zunächst mit den durch ein optimales Preis/Leistungsverhältnis bestechenden Datenspeichern der Firmen Frank & Britting sowie Erphi Electronic befassen, zumal man bei diesen Produzenten als Endabnehmer im Falle technischer Fragen anrufen darf, was bekanntlich bei Apple-München nicht erwünscht ist, obwohl man dort nicht verkennt, daß der Apple-Händler bei der Anpassung einer Festplatte normalerweise überfordert ist, denn schließlich ist ein Händler ein Händler und kein Sy-



stemprogrammierer. Wenn dann nicht der Produzent oder wie hier der Peeker hilft, dann hilft eben keiner.

2. Wozu größere Datenspeicher?

Falls Sie bislang nur 35- oder 40-Spur-Laufwerke benutzt haben und nunmehr auf ein 160-Spur-Subsystem der Firma Erphi und/oder auf eine Festplatte der Firma Frank & Britting umzusteigen gedenken, so sollten Sie folgende Punkte berücksichtigen. Vorab jedoch einige Definitionen:

Disk-II-Laufwerk: 35-Spur-Laufwerk für 5,25-Zoll-Disketten (SS/SD = single sided, single density = einseitig, einfache Spurdichte) mit 140K Speicherkapazität (= Netto-Speicherkapazität) pro Diskette nach Formatierung (1K = 1024 Zeichen); pro Controller 2 Laufwerke anschließbar (Controller = Steckkarte zur Datenübertragung zwischen Laufwerk und Rechner); Duodisk mit eigenem Controller als spezielles Doppellaufwerk erhältlich; Datenübertragungsrate normalerweise maximal 8,2K/s (gut 8.000 Zeichen/Sekunde); Bezugsquelle: Firma Apple; Preis für Doppellaufwerk inkl. Controller ca. DM 1400,- bis 1800,-, je nach Händler.

Erphi-Subsystem: 160-Spur-Doppellaufwerk für 5,25-Zoll-Disketten (DD/DD = double sided, double density) mit 640K Speicherkapazität pro Diskette; pro Controller 1 Doppellaufwerk oder auch 2 einzelne Fremdlaufwerke anschließbar; Datenübertragungsrate normalerweise maximal 8K/s; Bezugsquelle: Erphi-Electronic; Preis für Doppellaufwerk (FSS-280-Subsystem) inkl. Controller ca. DM 1700,-. Der Controller namens AFDC (= Autopatch Floppy Disk Controller) kann separat bezogen werden und auch zum Anschluß von 35- und 40-Spur-Laufwerken (z.B. Erphi F122) verwendet werden. In diesem Beitrag befassen wir uns jedoch nur mit der Komplettlösung = Erphi-FSS-280-Subsystem + Controller, da die Anpassung an von dritter Seite bezogene Laufwerke problematisch sein kann.

Megaboard-Festplatte: Harddisk-Controller und Festplatte; wahlweise 10 oder 20 Megabytes unformatiert; formatiert bei z.B. 10M = etwa 9608K netto (statt 10240K brutto); lieferbar als *interne* Megacore-Festplatte (Apple-Netzteil wird ausgetauscht) oder als *externe* MDB = Mobile Datenbox; Datenübertragungsrate normalerweise maximal 44K/s; Bezugsquelle: Frank & Britting; Preis für z.B. 10M-MDB inkl. Controller ca. DM 2700,- (für Megacore ca. DM 1000,- mehr). Der Controller (Megaboard i.e.S.) kann auch separat bezogen werden, doch ist hier noch mehr als beim Erphi-Controller vom Anschluß von Datenspeichern, die von dritter Seite erworben wurden, abzuraten, da sonst nicht nur bei technisch Unkundigen Anpassungsprobleme entstehen können. Über den Hüthig Software Service ist deshalb eine preiswerte Megaboard-Komplettlösung einschließlich eines angepaßten Dateiverwaltungsprogramms erhältlich.

2.1. Speicherkapazität

Ein Disk-II-Doppellaufwerk mit 2 · 140K Speicherkapazität reicht für viele Aufgaben völlig aus. Ein gutes Beispiel ist die Textverarbeitung:
Selbst wenn Sie pro Monat 1 Buch schreiben würden, so würden Sie doch nicht mehr als 4 Disketten pro Buch benötigen. Und wer schreibt schon eine halbe Million Zeichen pro Monat? Dies wären nämlich mehr als 22.000 Zeichen = 22 DIN-A4-Seiten (anderthalbzeilig) pro Werktag. Vom „Diskjockey" kann also hier keine Rede sein. Man beachte, daß wir hier bewußt von einem Doppellaufwerk gespro-

TurtleGraphics-Library-Paket

von Dieter Geiß

Turtle-Utilities für Fenstertechnik und Apple-Maus in einfacher und doppelter Hires-Grafik für Pascal 1.2 auf Apple IIe/c mit Maus oder Joystick. 2 Disketten mit umfangreichem Manual, DM 98,–. Unter Pascal 1.1 mit 64K nur eingeschränkt lauffähig

Im einzelnen bietet das Paket folgende Möglichkeiten:
- volle Kompatibilität mit der alten „TurtleGraphics"
- alle zeitkritischen Funktionen in reinem Assembler programmiert
- Benutzung der zweiten Hires-Seite ($4000-$5FFF) möglich
- für Apple IIc und Apple IIe mit erweiterter 80-Zeichen-Karte Benutzung der doppelten Hires-Grafik mit 560 × 192 Punkten bzw. 16 neuen Farben möglich
- schnelle Prozeduren zum Zeichnen eines Punktes oder einer Linie
- Benutzung mehrerer Zeichensätze gleichzeitig
- Scrolling des Hires-Schirms oder eines Teils in vier Richtungen
- drei verschiedene Schriftarten: Fett-, Breit- und Proportionalschrift, beliebig mischbar (acht Möglichkeiten)
- spezielle schnelle Ausgabe von Text
- Cursor bei Eingabe frei programmierbar
- Ein-/Ausgabe von INTEGER-, CHAR-, STRING- und REAL-Werten im Grafikmodus
- Menüzeile wie beim Macintosh
- Pull-down-Menüs
- Laden und Speichern von Fenstern (Windows)
- Öffnen von Fenstern
- Aktivieren und Deaktivieren von Fenstern
- Verschieben und Vergrößern/Verkleinern von Fenstern
- Scrolling von Fensterinhalten in allen vier Richtungen
- Umfangreiche Demos als Quelltexte.

Hüthig Software Service · Postfach 10 28 69 · 6900 Heidelberg

DISK40

Disketten-Organisationsprogramm für Apple II+, IIe oder IIc

von Hermann Seibold und Dipl.-Ing. Udo Marin, 1986, Programmdiskette mit Anleitung, DM 48,–

DISK40 entstand aus der Analyse bestehender Kopierprogramme und vereint in sich eine Vielzahl von Möglichkeiten, die sich als nützlich erwiesen haben. Durch eine einfach zu bedienende Menüführung können DOS-3.3-Disketten umfangreich bearbeitet oder kopiert werden.
Zu den vielfältigen Möglichkeiten des Programms zählen u. a.:
- Tabellarische Ausgabe der Diskettenbelegung
- Ordnen des Catalogs
- „Undelete"n von versehentlich gelöschten Dateien
- Vergleichen von Disketten, Dateien oder der DOS-Spuren
- Kopieren von Disketten, Dateien oder DOS-Spuren
- Formatieren von Daten-Disketten
- Erweitern auf 40 Spuren bei bestehenden 35-Spur-Disketten
- Ändern des Boot-Programms
- File-Editor zum Editieren von Disketten-Dateien
- Komfortabler Sektor-Editor für Hex- und ASCII-Darstellung
- VTOC-Editor, z. B. zur Freigabe der DOS-Spuren

Schon nach wenigen Minuten können, dank der ausführlichen Beschreibung, Disketten nach eigenen Wünschen modifiziert oder Daten nach einem Disk-Crash wieder gerettet werden.

Hüthig Software Service · Postfach 102869 · Heidelberg 1

chen haben, denn bei einem Einzellaufwerk ist die Datensicherung umständlich, zeitraubend und fehlerträchtig.

Es gibt jedoch viele Aufgaben, die sich mit 140K-Laufwerken nur begrenzt bewältigen lassen. Ein gutes Beispiel ist die Dateiverwaltung:
Unterstellen wir einen Datensatz von 200 Zeichen für die Summe aller Eingabefelder, so passen auf eine 140K-Datendiskette nur etwa 630 Datensätze. Wer größere Datenmengen zu verwalten hat, ist mit einer Festplatte oder einem größeren Diskettenlaufwerk erheblich besser bedient. Beim Erphi-Subsystem (= 160-Spur-Doppellaufwerk = 2 · 640K) würden auf eine einzige 640K-Diskette etwa 3.000 Datensätze passen, während eine 10-Megabyte-Festplatte etwa 50.000 Datensätze aufnehmen könnte.

2.2. Übertragungsrate

Neben der Speicherkapazität spielt die Datenübertragungsrate eine wichtige Rolle. Wir beschränken uns hier aus Platzgründen auf ProDOS:
Bei den Disk-II-Laufwerken beträgt die Übertragungsrate auf Blockebene für Lesen und Schreiben jeweils ca. 8,2K/s.
Beim Erphi-Subsystem muß man zwischen Lesen und Schreiben differenzieren. Auf Blockebene wird mit ca. 9,3K/s gelesen und mit ca. 6,8K/s geschrieben; Mittelwert ca. 7,9K/s.

Bei der Megaboard-MDB-Festplatte beträgt die Übertragungsrate unter ProDOS einheitlich ca. 42,7K/s für Blocklesen und Blockschreiben (bei DOS 3.3 übrigens mit 43,75K/s größer, siehe Abschnitt 5). Zum Vergleich soll die amerikanische Festplatte namens „Sider" eine Übertragungsrate von 15K/s haben (s. „Apple's Sonderheft", 1986, letzte Seite). Für die Apple-Festplatte namens „Profile" soll angeblich eine Übertragungsrate von 19K/s gelten. Damit ist die MDB-Festplatte gut 5mal schneller als ein Disk-II- oder ein Erphi-Drive und gut 3mal schneller als die Sider bzw. gut 2mal schneller als die Profile. Die MDB-Festplatte ist damit etwa so schnell wie eine gute RAM-Disk, bei der mit 35-55K/s (sehr stark von den „Softswitches" und dem „Mapping" abhängig) gerechnet werden kann.

Hinweis: Für die Zeittests wurde das Programm BAD.BLOCKS aus „Apple ProDOS für Aufsteiger", Bd. 2, S. 177 verwendet. Es sei ferner darauf hingewiesen, daß entweder der Megaboard-Driver seit dem Megacore-Test im Peeker, Heft 7/85, S. 76-77, ganz beachtlich verbessert wurde oder die mir z.Zt. ausgeliehene externe MDB schneller ist als die interne Megacore. Eine erneute Durchführung der Textfile-Tests (Heft 7/85, S. 77) ergab für den WRITE.TEST 128s statt 255s und für den READ.TEST 47s statt 65s.

Ein gutes Maß für das Gespann Megaboard- und Erphi-Controller mag auch das weiter unten beschriebene Backup-Programm PRODOS.BACKUP sein. Für die Datensicherung eines 3,5-Megabyte-MDB-ProDOS-Volumes mit Hilfe von 6 Erphi-160-Spur-Disketten benötigt man in der Richtung
(a) MDB → Erphi: 11min 5s (665s)
(b) Erphi → MDB: 8min, 40s (520s)
Bei (a) wird von der MDB-Festplatte gelesen und auf die Erphi-Disketten geschrieben, was sichtlich länger dauert als die umgekehrte Richtung.
Insgesamt müssen 3,5M gelesen *und* geschrieben werden. Die Datenübertragungsrate bezieht sich somit auf 7M. Das ergibt
(7M = 7.340.032 : 1024 = 7168K):
7.168K : 665s = 10,8K/s
7.168K : 520s = 13.8K/s
14.336K : 1.185s = 12,1K/s
Zum Vergleich sei hier erwähnt, daß sich mit dem ebenfalls weiter unten beschriebenen Programm KOPY.160.SPUR in 2min, 40s (= 160s) eine 160-Spur-Diskette (= 640K · 2 = 1280K) auf dem Subsystem überspielen läßt. Dies sind
1280K : 160s = 8K/s.

Wozu sind nun diese Übertragungsraten gut? Wenn man ein Programm morgens einmal startet, spielt es keine Rolle, ob es z.B. in 2s (MDB-Festplatte) oder in 10s (Erphi-Subsystem, Faktor 5) im Speicher ist. Wenn jedoch etwa bei einem Dateiverwaltungsprogramm größere Datenbestände durchsucht werden müssen, können ganz ernorme Wartezeiten entstehen. Dann ist es eben nicht mehr gleichgültig, ob man 30s oder 150s auf die Mattscheibe starrt. Die Kluft kann jedoch bei dateiintensiven Programmen noch erheblich größer werden, wenn nicht auf der bislang ausschließlich betrachteten Blockebene auf Dateien zugegriffen wird. Bei ungünstigem Motor-an/aus-Timing (s. „RWTS mit Warteschleife" in „Apple DOS 3.3", S. 116) sackt nämlich die Übertragungsrate bei Disketten dramatisch ab. Dies ist bei einer Festplatte niemals der Fall, denn diese rotiert ständig.

Technisches

Es ist zwar bei der MDB wie auch bei anderen Festplatten nicht irrelevant, auf welchen physikalischen Spuren sich die gewünschten Sektoren oder Blöcke befinden, doch bewegt sich diese physikalische Zugriffszeit in einem Bereich, der bei unserer Erörterung vernachlässigbar ist. Wie man beispielsweise dem Aufsatz „Aufbau und Funktion von Diskettensystemen" von G.Berg (Peeker, Heft 3/85, S.10-21) entnehmen kann, wird die sog. Block-Übertragungsrate in KBit/s gemessen. Beim Erphi-Controller sind dies
250 KBit/s → entspricht etwa
250.000 Bits : 8 : 1024 = 30K/s
und beim Megaboard-Controller
5 MBit/s. → entspricht etwa
5.000.000 Bits : 8 : 1024 = 600K/s.
Diese physikalischen Werte sind indes völlig utopisch, weil ein 8-Bit-Prozessor derartige Übertragungsraten nicht mitmacht. Das denkbar schnellste Kopierprogramm für das Erphi-Subsystem, nämlich „Superquick" von Arne Schäpers, bringt es auf eine praktische Übertragungsrate von ca. 14 bis 15K/s (ohne Verify).

Faustregeln

– Das Erphi-Subsystem ist dann vorzuziehen, wenn die Speicherkapazität den gestellten Anforderungen entspricht und eine hohe Datenübertragungsrate nicht benötigt, also nicht ständig mit dateiintensiven Programmen gearbeitet wird.
– Umgekehrt ist das Megaboard-System dann vorzuziehen, wenn eine noch größere Speicherkapazität erforderlich ist und/oder eine hohe Datenübertragungsrate bei Dateiverwaltungsprogrammen benötigt wird.
Diese Faustregeln heben auf den professionellen Einsatz ab. Es sei nicht verschwiegen, daß ein (Hobby)-Programmierer, falls er sich eine MDB leisten kann, ein ganz neues „Programmier-Feeling" erlebt, denn alle vier Betriebssysteme (DOS 3.3, ProDOS, CP/M und UCSD-Pascal) können direkt von der Festplatte gebootet werden, und die Assemblier- und Compilierzeiten werden geradezu dramatisch reduziert. Auch das beim Programmieren häufig erforderlich Neubooten ist bei der MDB eine Angelegenheit von etwa 2-3s, wobei der Controller intelligent genug ist und automatisch das zuletzt verwendete Betriebssystem startet (und nicht wie beim Kaltstart DOS 3.3).

3. Praxis-Tips

Gegenüber den alten 35-Spur-Laufwerken muß der Umgang mit dem Erphi-Subsystem und der Megaboard-Festplatte erst erlernt werden. Abgesehen von der einmalig und etwa eine Stunde dauernden Konfigurierung der Betriebssysteme, auf die wir hier aus Platzgründen nicht eingehen, sind einige Spielregeln und auch Einschränkungen zu beachten.

3.1. Slotabhängigkeit

Wenn Sie einmal Pascal oder CP/M von z.B. Slot 5 gebootet haben sollten, so werden Sie der Meldung „Must boot from slot 6" begegnet sein. Nur die Betriebssyste-

INTUS-Lern- und Anwenderprogramme

für Apple IIe/IIc - Computer

- Maschinenschreiben wie der Blitz DM 175,–
- Business-English, 2800 Begriffe, 16 Bereiche, 3 Übungsarten DM 120,–
- Rechtschreibtrainer, 4000 Wörter, 16 Bereiche, 4 Übungsarten DM 120,–
- Basic-Lernprogramm, ausgezeichnet DM 238,–
- Kinderschule, für Vorschulkinder DM 49,–
- Rechenmodelle für AppleWorks DM 195,–
- MailWorks für AppleWorks-Serienbriefe, für alle Drucker geeignet DM 168,–
- AppleGraph, Erstellen von Balken- und Kreis-Graphiken DM 120,–
- PriBu-Privatbuchhaltung DM 195,–
- DMP-Charger zur Gestaltung eigener Zeichensätze auf dem Matrix-Drucker DM 198,–
- Gesamtkatalog mit über 200 Programmen gratis
- **Demo-Diskette gratis** gegen Angabe/Einsendung dieses Inserates
- 6000 Frei-Programme (fast) gratis (Shareware/Public domain) Liste DM 10,–

INTUS SOFTWARE

Kaiserstr. 21, 7890 Waldshut,
Tel. 0 77 51-79 20

Ausgabe und **Eingabe** mit

TYPETERM®

im Slot Ihres

APPLE II/IIe

Das bedeutet: Computer-textverarbeitung von der Schreibmaschinentastatur! Steckerfertig ohne Umbau.

Die neue CE-550!
mit TYPETERM DM 1.398,–

TYPETERM- **DM 479,–**
Interface
für alle BROTHER-Typenrad-schreibmaschinen ab CE-51 bis EM-250

CE-68 mit TYPETERM ..... DM 1947,–
EM-80 mit TYPETERM ..... DM 1887,–
TYPETERM-Kit für CE-50 ..... DM 468,–
Cable Kit A
(erforderlich ab EM-80) ..... DM 103,–

TYPETERM – ein starkes Interface für starke Maschinen! Alle Cursor- und Ctl-Befehle. 4k ROM auf der Karte für DOS, PRODOS, CP/M, PASCAL. 2 Zeichensätze verfügbar z. B. deutsch u. ASCII. Alle Features: Hoch-/Tiefstellen, autom. Unterstreichen, var. Zeichen und Zeilenabst., autom. Papierzuführung usw.

TYPETERM – ein Produkt von

interkom electronic Kock & Mreches GmbH Postf., 3004 Isernhagen 4 Telefon 0 51 39-8 73 93

Ausgabe mit

TYPETERM® JUNIOR

im Slot Ihres

APPLE II/IIe

Paketpreis **DM 899,–**
Schreibmaschine AX-10 mit Interface TYPETERM JUNIOR, steckfertig.

CE-550

brother
Die Zukunft heute

TYPETERM JUNIOR mit AX-10 – unser besonders günstiges Gespann, ebenfalls steckfertig. Mit TYPETERM JUNIOR kann die AX-10 mehr. Sie wird zum vollwertigen Typenraddrucker für Ihren Apple:

- 3 verschiedene Schriftstärken
- Automatisches Unterstreichen
- 2 Zeichensätze z.B. deutsch u. ASCII
- 2 Zeichenabstände
- 2k ROM auf der Karte für Ausgabe unter DOS, PRODOS, CP/M u. PASCAL.

TYPETERM JUNIOR – ein Produkt von

interkom electronic Kock & Mreches GmbH Postf., 3004 Isernhagen 4 Telefon 0 51 39-8 73 93

ZUSATZ-KARTEN:

V-24-Schnittstelle	199,-	**Z-80-Karte**	98,-
80-Zeichen-Karte m.Softswitch	236,-	**16 K-Language-Karte**	98,-
Joy Stick De Luxe	59,-	**Accelerator** 3,6 MHz	950,-
68000 Intemex	1600,-	**PAL Karte**	110,-
RGB Karte	239,-	**IEEE 488**	312,-
Koppler dataphon m. FTZ	325,-	**Z 80 B Karte** mit Software	919,-
Centronics-Karte von Epson für Graphik	210,-	für Text	145,-
Centronics-Schnittstelle für 2 Drucker gleichzeitig			129,-

Super-Eprommer

belegt keinen Slot, incl. Software für 2716-27128 ..... **239,-**

Floppy-Controller

FDC 4 für alle Laufwerke ..... 169,- Bausatz wie links 159,-
Leerplatine wie oben incl. Prom u. Eprom 98,-

Erphi-Controller 298,-

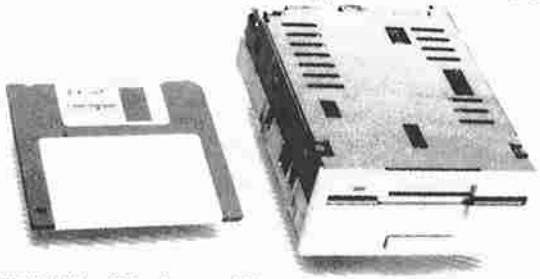
Disketten 1D, 48 tpi ..... 10 St. **29,-**
Disketten 2D, 48 tpi ..... 10 St. **36,-**

Frei programmierbare Keyboards

Wir bieten Ihnen die **Preh-Qualität** auch für Apple. AK 87 spez. mit Gehäuse, Anschlußkabel, Zehner-Tastenfeld, dt. Zeichensatz, Sondertasten für Ctrl-Codes und Rechenfunktionen **339,-**
Preh Commander Keyboard, frei programmierbar bis zu 10 Ebenen, pro Taste bis zu 250 Zeichen ..... nur **599,-**
Gleiche Tastatur wie oben **für Apple IIe** ..... nur **698,-**

TEAC 3½" Laufwerk FD 35 F 498,-

Speicherkapazität 1 MB, (formatiert 640 KB) jetzt für nur .....



TEAC FD 55 AV 1 x 40 Track	395,-	**TEAC FD 55 BV** 2 x 40 Track	460,-
TEAC FD 55 EV 1 x 80 Track	445,-	**TEAC FD 55 FV** 2 x 80 Track	398,-

Apple®-kompatibles Laufwerk incl. Gehäuse + Kabel 599,-

320 KB Laufwerk für IIc **948,-**
640 KB Laufwerk für IIc **1088,-**

Panasonic Drucker: **1090** ..... nur **849,-**
1091 ..... nur **1095,-** **1092** ..... nur **1295,-**

Die Microfloppy mit Zukunft:

Speicherkapazität: 2 x 1 MByte formatiert: 2 x 640 kByte. Anschlußfertig mit PROM-residenter Patchsoftware für CP/M 2.2, Apple DOS 3.3, DiversiDOS 2-C, 4-C (DD MOVER), Apple Pascal 1.1, Pascal 1.2, Pro-DOS 1.0.1, 1.1, 1.1.1 zum Preis von **1498,-**
Low Power Version 1598,-

Mega-Core

mit Software für DOS 3.3, CP/M 2.20, Pascal, Pro-DOS, von Frank & Britting, 20 MB **4650,-**

Sonderangebot

Distar Laufwerk für II + IIe,

incl. Kabel u. Gehäuse ..... jetzt nur **349,-**

Gesamt-Preisliste anfordern! **Preise inclusive gesetzlicher Mehrwertsteuer.**
Händlerpreisliste bitte schriftlich anfordern!

UEDING electronics

Holtewiese 2
5750 Menden 1

DFÜ 02373/66877
Tel. 02373/63159

me DOS 3.3 und ProDOS sind prinzipiell slotunabhängig. Das Erphi-Subsystem sowie die Megaboard-Festplatte sind teils weniger und teils mehr slotabhängig.
Der Megaboard-Controller kann *allein* theoretisch in jeden Slot (außer Slot 0 beim II+ oder Slot 3 beim IIe) gesteckt werden. Normalerweise wird jedoch Slot 7 empfohlen, von dem aus dann alle vier Betriebssysteme einschließlich CP/M und Pascal direkt von der Platte gebootet werden können.
Der Erphi-Controller sollte *allein* normalerweise in Slot 6 gesteckt werden, doch sind unter DOS 3.3 und ProDOS auch Slot 5 und 4 zulässig. Allerdings funktioniert das mitgelieferte Kopierprogramm ECOPY beispielsweise nicht in Slot 5.
Wenn Megaboard- *und* Erphi-Controller *gleichzeitig* benutzt werden sollen, so muß der Megaboard-Controller in Slot 7 und der Erphi-Controller in Slot 6 gesteckt werden. Andernfalls würde nach dem Booten der Festplatte von Slot 7 die zusätzlich für das Erphi-Subsystem gedachte Betriebssystem-Modifikation nicht ausgeführt werden. Man darf also dann nicht mit PR#6 vom Erphi-Subsystem booten, sondern muß vielmehr stets mit PR#7 (oder Ctrl-Apfel-Reset) die Festplatte aktivieren.

3.2. Disketten und Kopierschutz

Ein wesentlicher Vorzug der Disk-II-Laufwerke bestand darin, daß selbst bei billigen SS/SD-Disketten und selbst bei nachlässigem Umgang relativ selten Lesefehler auftraten. Beim Erphi-Subsystem sollte man jedoch nicht nur DS/DD-Disketten verwenden – z.B. die Scientific-Disketten von BASF, von denen wir 100 Exemplare getestet haben, ohne daß irgendwelche Fehler aufgetreten wären –, sondern zudem mit den Disketten sorgfältiger umgehen. Zudem ist das Mischen von 35- bzw. 40-Spur- sowie 160-Spur-Disketten grundsätzlich nicht möglich. Das nachfolgende Schaubild mag dies verdeutlichen:

Anfang einer 80-Spur-Diskettenseite

```
 Ø,265  Ø,265  Ø,265  Ø,265      Spurabstand
|------|------|------|------|-
       Ø      1      2      3   Spur-Nr.
|------**------**------**------**-
   Ø,15   Ø,15   Ø,15   Ø,15     Spurbreite
```

Anfang einer 40-Spur-Diskettenseite

```
 Ø,53Ø          Ø,53Ø           Spurabstand
|--------------|--------------|-
       Ø              1        Spur-Nr.
|-----****------------****------------
   Ø,3Ø           Ø,3Ø          Spurbreite
```

Die 80 konzentrischen Spuren einer Seite einer 160-Spur-Diskette haben einen Abstand (symbolisiert durch |) von der vorangehenden zur nächsten Spur, der etwa 0,265mm beträgt (sog. *Spurabstand*). Wenn 1 Zoll (= Inch) 2,54cm bzw. 25,4mm entspricht, so würden 96 Spuren exakt 1 Zoll einnehmen (0,265 · 96 = 25,4mm = 1 Zoll). Man spricht deshalb auch von 96 TPI (= Tracks per inch = Spuren pro Zoll). (Da die 160-Spur-Diskette jedoch pro Seite nur 80 Spuren hat, ist die Gesamtlesezone übrigens nur ca. 2cm breit.) Demgegenüber ist der Spurabstand bei einer 40-Spur-Diskette doppelt so groß, nämlich ca. 0,53mm, womit eine 48-TPI-Diskette vorliegt (0,53 · 48 = 25,4mm = 1 Zoll).

Dem Diagramm kann man weiterhin entnehmen, daß die *Spurbreite* bei der einen Seite einer 160-Spur-Diskette (ca. 0,15mm) nur die Hälfte der Spurbreite einer 40-Spur-Diskette (ca. 0,30mm) einnimmt. Der Lese/Schreibkopf eines 160-Spur-Laufwerks muß also halb so breit sein wie der Kopf eines 40-Spur-Drives.
Aus der Abbildung ist weiterhin ersichtlich, daß die Spuren 0, 1, 2 usw. einer 40-Spur-Diskette, wenn man die Spurbreite ignoriert, dieselben Positionen einnehmen wie die Spuren 0, 2, 4 usw. der einen Seite einer 160-Spur-Diskette, und zwar der Seite 0, die sich unten bzw. auf der Nicht-Etikettseite befindet. Die Etikettseite selbst heißt Seite 1.

Daraus erhellt nunmehr im einzelnen:
– Eine auf dem Erphi-Subsystem mit PR#6 (= 160-Spur-Modus) gebootete 35-Spur-Diskette, die auf einem Disk-II-Laufwerk formatiert wurde, läßt sich nicht lesen, geschweige denn beschreiben, weil die Spuren verwechselt werden. Statt der Spur 1 liest der Controller von der Mitte zwischen Spur 0 und 1, und dort ist „nichts".
– Eine auf dem Erphi-Subsystem mit IN#6 (= 35- bzw. 40-Spur-Modus) gebootete 35-Spur-Diskette, die auf einem Disk-II-Laufwerk formatiert wurde, läßt sich *lesen*, doch darf diese Diskette nicht vom Subsystem beschrieben werden, weil dann zu schmale Spuren entstehen würden, die dann wiederum vom dem Disk-II-System nicht mehr gelesen werden könnten, weil dessen breiterer Lesekopf dann neben der schmalen Spur noch die „Reste" der ursprünglichen breiteren Spur lesen würde, und dies führt unweiterlich zu einem I/O-Error.
– Nach dem Booten einer 35-Spur-Diskette auf dem Erphi-Subsystem mit IN#6 läßt sich auf dem Subsystem eine andere (Leer)diskette formatieren und danach im 35- bzw. 40-Spur-Modus vom Subsystem lesen *und* beschreiben. Doch kann diese Diskette dann trotzdem nicht mehr von einem Disk-II-Drive gelesen werden, obwohl man eigentlich annehmen müßte, daß ein breiter Lesekopf auch eine schmale Spur lesen kann.

Faustregeln
1. Das Lesen von einer 35-Spur-Diskette ist mit dem Erphi-Subsystem grundsätzlich möglich, wenn mit IN#6, d.h. im 40-Spur-Emulationsmodus, gebootet wird. Deshalb können auch die meisten kopiergeschützten Disketten vom Subsystem gestartet werden. Programmdisketten, die jedoch als Bestandteil des Kopierschutzes auf die Diskette zurückschreiben, dürften sich folgerichtig insofern selbst zerstören, als sie sich später von einem Disk-II-Drive nicht mehr selbst lesen können.
2. Wenn man für jedermann lesbare 35-Spur-Disketten herstellen will, so muß man diese auf einem 35-Spur-Laufwerk duplizieren. Obwohl beispielsweise unser Sammeldisketten-Kopierer zusätzlich ein Erphi-Subsystem besitzt, könnte er es aus diesem Grund niemals zur Herstellung der Sammeldisketten benutzen.

Während beim Erphi-Subsystem kopiergeschützte Programme nur mit den erwähnten Einschränkungen benutzt werden können, ist bei der Megaboard-Festplatte die Verwendung solcher Programme in der Regel nicht möglich, denn der im „Ressort Paranoia" behandelte Verfolgungswahn namens Kopierschutz findet immer Mittel und Wege, um ein sinnvolles Diskettensystem außer Kraft zu setzen. Beispielsweise ist der DOS-3.3-Applewriter kopiergeschützt und läuft deshalb nicht auf Festplatten, doch kann man, was nicht in der Anleitung steht, beim Programmstart auf die „C"-Taste drücken, worauf (auch beim deutschen Applewriter) am Bildschirm die englischen Meldung „Do you have Corvus y/n" erscheint. Tippt man nun „y", so stürzt der Applewriter auch dann ab, wenn man eine Corvus-Festplatte besitzt, wie mir zwei Corvus-Besitzer bestätigt haben. Hier hat sich wohl ein paranoides Gemüt gedacht: „Denen werde ich jetzt mal zeigen, was mein Kopierschutz alles kann!" Fazit: Benutze niemals kopiergeschützte Programme, wenn du nicht selbst paranoid werden willst!

3.3. Patch-Kompatibilität

Abgesehen von der slotmäßigen Einschränkung, dem Erfordernis eines pfleglichen Umgangs mit 160-Spur-Disketten, der nur bedingten Benutzbarkeit von 35-Spur-Disketten auf dem 160-Spur-Subsystem sowie dem Kopierschutzproblem ist auch auf das Auto-Patch-Verfahren einzugehen. Erfreulicherweise muß man hier

konstatieren, daß sich erstens die Patches auf ein absolutes Minimum beschränken und daß zweitens Erphi-Subsystem und Megaboard-Festplatte fast reibungslos miteinander kommunizieren. Ein kleiner Wermutstropfen ist nämlich die CP/M-Version 2.26 von Microsoft, die zwar jetzt auch auf dem Erphi-Subsystem funktioniert, nicht jedoch auf der MDB- oder Megacore-Festplatte, die bislang nur CP/M 2.2 unterstützt. Im einzelnen können folgende Betriebssysteme gefahren werden:
– DOS 3.3, Diversi-DOS 2C und 4C
– Pascal 1.1, 1.2 (auch 128K-Version)
– ProDOS 1.0.1, 1.0.2, 1.1.1
– CP/M 2.2 sowie zusätzlich bei Erphi 2.20B, 2.23 und 2.26
Ein Leser schrieb anläßlich des Beitrages „Ein intelligenter Harddisk-Controller" (Peeker, Heft 4/86, S. 20ff.), daß er Angaben darüber vermisse, ob beispielsweise der TASC-Compiler auf der Megaboard-Festplatte funktioniere. Dies kann man bejahen, denn der Megaboard-Patch beschränkt sich bei DOS 3.3 auf 3 Bytes bei $BD00. Durch den Erphi-Controller werden jedoch noch weitergehende Patches vorgenommen, wodurch das Doppellaufwerk in vier virtuelle Drives D1, D2, D3 und D4 eingeteilt wird, von denen ein Anwenderprogramm in der Regel nichts weiß.

4. KOPY.160.SPUR

Wenn man sich zum Kauf eines Massenspeichers entschließt, so muß man sich auch Gedanken über die Datensicherung machen. Bei dem Erphi-Doppellaufwerk ist das Duplizieren von Disketten ebenso unkompliziert wie bei den Disk-II-Drives, doch dauert das Kopieren einer 640K-Diskette natürlich erheblich länger als bei einer 140K-Diskette. Das von Erphi Electronic mitgelieferte sog. ECOPY ist leider alles andere als schnell und formatiert im übrigen nicht das Duplikat, so daß die Formatierung gesondert vorgenommen werden muß. Das weiter unten gelistete Programm KOPY.160.SPUR formatiert zwar ebenfalls nicht, ist jedoch bereits spürbar schneller als ECOPY. Am schnellsten ist jedoch das Programm SUPERQUICK des Hüthig Software Service, zumal es zusätzlich formatiert und außerdem noch einen Verify beinhaltet. Für eine 160-Spur-Diskette ergeben sich bei den drei Programmen folgende Kopierzeiten:

ECOPY: 3min, 25s, ohne Format
KOPY.160.SPUR: 2min, 40s, ohne Format
SUPERQUICK: 1min, 53s, mit Format

Da KOPY.160.SPUR dem Programm PRODOS.BACKUP weitgehend ähnlich ist, beschränken wir uns hier auf die Handhabung:

1 In dem Applesoft-Programm KOPY.160.SPUR können Sie in der Zeile 110 die Konstanten DU = Originalslot und HU = Duplikatslot ändern. Die Voreinstellung
110 DU = 6: HU = 6
besagt, daß von Slot 6, Drive 1 nach Slot 6, Drive 2 kopiert wird. Wenn Sie Ihr Erphi-Subsystem beispielsweise in Slot 5 installiert hätten, so mußte es lauten:
110 DU = 5: HU = 5
Wenn Sie zusätzlich die Drives ändern möchten, was allerdings nur erforderlich wäre, wenn Sie über 2 Subsysteme verfügen würden, so können Sie die Zeilen 170 und 180 modifizieren.
2 Das Assemblerprogramm KOPY.160.SPUR.O ändert die ProDOS-System-Bitmap in der Form, die aus dem Listing von PRO.BACKUP.O, Zeilen 92-100, ersichtlich ist, und benutzt damit den Speicherbereich $0A00-$BE00 als Kopierpuffer.
(a) Starten Sie also unter ProDOS das Programm mit
RUN KOPY.160.SPUR
(b) Legen Sie in Drive 1 das Original und in Drive 2 das spätere Duplikat und beginnen Sie den Kopiervorgang nach „W = Weiter".
(c) Falls weitere Kopien erforderlich sind, so können Sie am Ende der Kopie automatisch zu Schritt (b) zurück.
(d) Wenn Sie das Programm verlassen haben, so müssen Sie mit z.B.
PR#6
neu booten, weil der Kopierpuffer das BASIC.SYSTEM zerstört hat.
Auch wenn KOPY.160.SPUR unter ProDOS gestartet wird, können trotzdem auch Disketten der anderen Betriebssysteme (DOS 3.3, Pascal, CP/M) dupliziert werden, die jedoch in jedem Fall bereits vorformatiert sein müssen.

Übrigens kann KOPY.160.SPUR auch bei 35-, 40- und 80-Spur-Drives verwendet werden. In diesem Fall muß man in der Zeile
120 DB = 1280: HB = 1280
die entsprechende Disketten-Blockgesamtanzahl bei beiden Variablen einsetzen. Für 35-Spur-Disketten würde gelten:
120 DB 280: HB = 280

5. PRODOS.BACKUP

5.1. Geschwindigkeitsvergleiche

Im Peeker, Heft 5/86, haben wir bereits ein DOS-3.3-Backup-Programm namens MDB.KOPY für 140K-DOS-Volumes der „mobilen Datenbox" vorgestellt, das mit einer Datenübertragungsrate von ziemlich genau 12,44K/s dem nachfolgenden ProDOS-Backup-Programm namens PRODOS.BACKUP für den MDB-FSS-280-Datentransfer mit 12,10K/s geringfügig überlegen ist. Trotzdem sind 12,1K/s – gewichtet aus ca. 42,7K/s für die Festplatte und ca. 8K/s für die Floppy – für einen 6502-Rechner ein ungewöhnlich hoher Wert. Denn für z.B. 3,5 Megabytes oder konkreter 3.666.944 Bytes benötigt man nämlich nur 9min bis 11min für die Datensicherung. Damit sind das DOS-3.3- und ProDOS-Backup-Problem bei der „mobilen Datenbox" gelöst.

Interessieren Sie die Gründe für das geringfügig schlechtere Abschneiden von ProDOS gegenüber DOS 3.3?
1. Bei PRODOS.BACKUP enthalten die im obigen Abschnitt „2.2. Übertragungsrate" genannten Zeiten, z.B. 8min, 40s für 3,5 Megabytes in Richtung vom Erphi-Subsystem zur Megaboard-MDB, bereits die für das Wechseln der 6 erforderlichen Disketten benötigten Sekunden.
2. Wie oben ebenfalls bereits erwähnt, ist bei einer Erphi-160-Spur-Diskette der Lesevorgang erheblich zügiger als der Schreibvorgang. Diese Unterschiede gelten bei Disk-II-Laufwerken nicht.
3. Auf der MDB-Festplatte werden bei der Konfigurierung zuerst die DOS-3.3-, dann die UCSD- und CP/M- und erst zum Schluß die ProDOS-Harddisk-Spuren formatiert. Möglicherweise ist deshalb die Übertragungsrate unter DOS 3.3 (43,7K/s) größer als unter ProDOS (42,7K/s). Vielleicht ist aber auch der ProDOS-Driver auf dem Controller weniger intelligent als der DOS-3.3-Driver geschrieben, denn wie mir selbst von eigenen RAM-Disk-Drivern bekannt ist, können einige unnötige 6502-Befehle bereits 1K/s ausmachen.

5.2. Das Pufferproblem

Während das Backup-Programm MDB.KOPY aus dem letzten Peeker im Prinzip ein modifiziertes und getuntes 140K-Kopierprogramm darstellt, muß man bei einem flexiblen ProDOS-Datensicherungsprogramm von variablen Speichergrößen und Puffern ausgehen. Im einzelnen muß man unterscheiden:
– Größe der Harddisk
– Größe der Floppy-Disks
– Größe des RAM-Puffers
Der Inhalt einer Harddisk entspricht in der Regel dem Inhalt mehrerer Floppy-Disks. Doch wird die letzte der Backup-Floppies meist nicht komplett gefüllt sein. Folglich muß man zwischen einer oder mehreren Komplettdisks sowie ggf. einer Restdisk unterscheiden. Nehmen wir fernerhin an, daß die Datenübertragung aus Gründen der Geschwindigkeit über einen größeren RAM-Puffer erfolgt. Dann muß für die Komplettdisks der RAM-Puffer mehrmals ganz sowie zum Schluß ggf. teils gefüllt werden. Für die Restdisk gilt das gleiche,

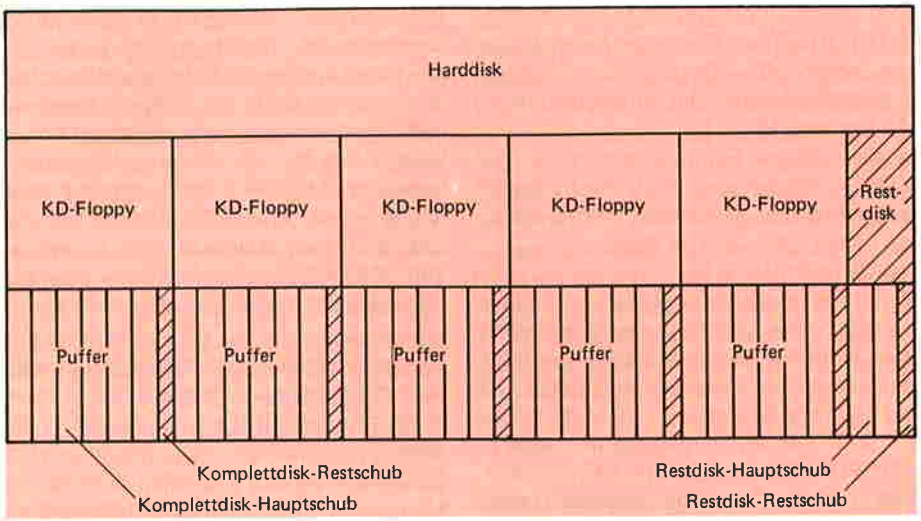


Abb. 1

wobei man jedoch zwischen dem Komplettdisk-Restpuffer und dem Restdisk-Restpuffer unterscheiden muß. Betrachten wir hierzu die **Abb. 1** und rechnen wir ein kleines Beispiel durch:

1 Der RAM-Puffer des Programms PRODOS.BACKUP kann theoretisch den Bereich $0A00-$BE00 einnehmen, doch legen wir hier aufgrund einer als weitgehend optimal ermittelte Größe den RAM-Puffer auf $1000-$9000 = 32K = 128 Pages = *64 Blöcke* fest.

2 Auf der Megaboard-Festplatte dürften regelmäßig mehrere Megabytes für ProDOS reserviert sein, wobei wir für unser Rechenbeispiel 3,5M = 7168 Blöcke (0-7167) annehmen. Block 0 der Harddisk wird aus dem nachstehend genannten Grund niemals kopiert, so daß sich 7168-1 = *7167 Blöcke* ergeben.

3 Auf eine 160-Spur-Diskette des Erphi-Subsystems passen 0,625M = 1280 Blökke (0-1279). Für die spätere Identifikation der Backup-Disketten reservieren wird jedoch jeweils den Block 0, der das Backup-Datum und die Disk-Nummer enthalten wird. Damit ergeben sich 1280-1 = *1279 Blöcke*. Es gibt hierbei noch einen Trick: Wenn wir bei der *ersten* Backup-Floppy mit Block 1 und ebenso bei der Harddisk mit Block 1 beginnen, so stimmt die Reihenfolge der Blöcke 1-1279 auf der Backup-Disk mit der der Harddisk überein, so daß wir anhand eines CATALOGs der *ersten* Backup-Disk später leicht identifizieren, was beim letzten Backup dupliziert wurde.

4 Nun teilen wir die Harddisk-Blockanzahl durch die Blockanzahl einer Komplettdisk. Das ergibt 7167 : 1279 = 5 Komplettdisks (5 · 1279 = 6395 Blöcke) + 1 Restdisk mit 772 Blöcken; 6395 + 772 = 7167 Harddisk-Blöcke.

5 Jetzt ermitteln wir für die 5 Komplettdisks (mit jeweils 1279 Blöcken) die Anzahl der Komplettdisk-Hauptschübe mit vollem RAM-Puffer ($1000-$9000 = 64 Blöcke) sowie das Pufferende beim Komplettdisk-Restschub, der ggf. den RAM-Puffer nicht voll ausnutzt. Das ergibt 1279 : 64 = 19 Komplettdisk-Hauptschübe (19 · 64 = 1216 Blöcke) sowie 1 Komplettdisk-Restschub ($1000-$8E00) mit 63 Blökken; 1216 + 63 = 1279 Blöcke pro Komplettdisk.

6 Schließlich ermitteln wir für die Restdisk (mit 772 Blöcken) die Restdisk-Hauptschübe mit vollem RAM-Puffer sowie den Restdisk-Restschub mit ggf. partiell gefülltem RAM-Puffer. Das ergibt 772 : 64 = 12 Restdisk-Hauptschübe (12 · 16 = 768) und 1 Restdisk-Restschub ($1000-$1800) mit 4 Blöcken; 768 + 4 = 772 Blöcke für die Restdisk.

5.3. Praktische Handhabung

1 Formatieren Sie zunächst eine Anzahl von Backup-Disketten und beschriften Sie die Labels mit den fortlaufenden Nummern 1, 2, 3 usw. sowie mit dem Backup-Datum in der Form TT.MM.JJ.

2 Wenn sich das Erphi-Subsystem in Slot 6 und die Megaboard-Harddisk in Slot 7 befinden, so booten Sie zunächst ProDOS und starten Sie dann mit
RUN PRODOS.BACKUP
das Applesoft-Programm von einer 160-Spur-ProDOS-Diskette. Bei einer anderen Slot-Belegung ändern Sie bitte vorher die Programmzeilen
120 DU = 6: REM Disk-Unit, S6, D1
130 HU = 7: REM Hard-Unit, S7, D1

3 Das Applesoft-Programm ermittelt nun die Floppy-Disk-Anzahl usw. und verlangt dann eine Backup-Richtungsangabe:
(1) Megaboard → Erphi
(0) Erphi → Megaboard

4 Für die Richtung Mega→Erphi wird das heutige Datum, für die Richtung Erphi→Mega das damalige Datum angefordert. Beschriften Sie die Backup-Disketten immer klar und deutlich, sonst werden Sie weder an diesem noch an anderen Backup-Programmen Freude haben!

5 Nach einer letzten, warnenden Bildschirmübersicht wird automatisch das Assemblerprogramm BACKUP.PRODOS.O geladen, das Sie dann jeweils auffordert, die Diskette Nr. X einzulegen. Bei der Datensicherung in Richtung Mega→Erphi erfolgt keine Prüfung, wohl aber in Richtung Erphi→Mega, wobei nur Disketten mit richtiger Nummer und richtigem Datum akzeptiert werden, so daß Verwechslungen ausgeschlossen sind.

Fortgeschrittene Programmierer können bei anderen Floppy-Disk-Laufwerken unterschiedliche Puffergrößen ausprobieren. Für das Erphi-Subsystem ergaben sich folgende Kopierzeiten in Richtung Mega-Erphi:
$0A00-$BE00 = 11min, 45s (Bitmap!)
$0A00-$9A00 = 11min, 25s
$1000-$9000 = 11min, 5s
Ein nicht allzu großer RAM-Puffer führte also zu den besten Übertragungsraten.

Kurzhinweise

1. Zweck:
(a) ProDOS-Backup-Programm PRODOS.BACKUP zur Datensicherung von Megaboard-Harddisk-Volumes; modifiziert auch bei anderen Festplatten verwendbar.
(b) ProDOS-Kopierprogramm KOPY.160.SPUR für Erphi-160-Spurlaufwerke; modifiziert auch bei kleineren Laufwerken verwendbar.
2. Konfiguration:
(a) Apple II+/e mit ProDOS und mit Megaboard-Festplatte in Slot 7 (MDB oder Megacore) und Erphi-160-Spur-Subsystem in Slot 6; bei anderer Slotbelegung PRODOS.BACKUP modifizieren.
(b) Apple II+/e mit ProDOS und mit Erphi-160-Spur-Subsystem in Slot 6.
3. Test:
(a) RUN PRODOS.BACKUP
Formatierte Disketten bereitlegen und zu Testzwecken die Richtung Mega→Erphi wählen.
(b) RUN KOPY.160.SPUR
Formatierte Duplikatdiskette bereitlegen.
4. Sammeldisk:
(a) ProDOS-Backup-Programm:
PRODOS.BACKUP (Applesoft)
T.PRO.BACKUP.O (Big-Mac-Quelltext)
PRO.BACKUP.O (Maschinenprogramm)
(b) 160-Spur-Kopierprogramm:
KOPY.160.SPUR (Applesoft)
T.KOPY.160.SPUR.O (Big-Mac-Quelltext)
KOPY.160.SPUR.O (Maschinenprogramm)
5. Sonstiges:
Die Programme müssen zunächst von der DOS-3.3-Sammeldisk mit DOSTOPRO oder CONVERT auf Ihre ProDOS-Arbeitsdiskette konvertiert werden.

PRODOS.BACKUP

```
120 DU = 6: REM DISK-UNIT,S6,D1
130 HU = 7: REM HARD-UNIT,S7,D1
150 PA = 04096: REM PUF-ANF:$1000
160 PE = 36864: REM PUF-END:$9000
170 PL = PE - PA: IF PL / 512 < > INT (PL / 512) THEN PRINT
    "NUR GANZE BLOCKANZAHL!": END
180 IF PEEK (49002) < > 0 AND PEEK (49003) < > 63 THEN PRINT
    "ZU WENIG SPEICHER!": END
190 P = 768: REM PAGE-3-PARAMETER
200 POKE P,DU * 16 + 0: REM +128-FUER-D2
210 POKE P + 1,HU * 16 + 0: REM +128-FUER-D2
220 PRINT CHR$ (21): TEXT : HOME : PRINT : INVERSE :
    HTAB 10: PRINT "** PRODOS.BACKUP **":
    HTAB 10: PRINT "** VON U. STIEHL **": NORMAL :
    PRINT CHR$ (4);"PREFIX": INPUT SP$: REM START-PREFIX
230 PRINT CHR$ (4);"PREFIX,S";DU;",D1": PRINT CHR$ (4);"PREFIX":
    INPUT P$: PRINT CHR$ (4);"BLOAD";P$;",TDIR,A$2000,L$100"
240 DB = PEEK (8233) + PEEK (8234) * 256:DB = DB - 1:
    POKE P + 3, INT (DB / 256): POKE P + 2,DB -
    PEEK (P + 3) * 256: REM $2029:DISK-BLOCKANZAHL-1
250 PRINT CHR$ (4);"PREFIX,S";HU;",D1": PRINT CHR$ (4);"PREFIX":
    INPUT P$: PRINT CHR$ (4);"BLOAD";P$;",TDIR,A$2000,L$100"
260 HB = PEEK (8233) + PEEK (8234) * 256:HB = HB - 1:
    POKE P + 5, INT (HB / 256): POKE P + 4,HB -
    PEEK (P + 5) * 256: REM $2029:HARD-BLOCKANZAHL
270 IF HB < 280 OR DB < 280 OR HB < = DB THEN HOME :
    PRINT "HARD-BLOCKS <= DISK-BLOCKS?": END
280 KD = INT (HB / DB): POKE P + 6,KD:RD = 0: POKE P + 7,RD:
    POKE P + 8,0: POKE P + 9,0: REM KOMPLETTDISKS
290 IF HB / DB > KD THEN RD = 1: POKE P + 7,RD:
    RB = HB - KD * DB: POKE P + 9, INT (RB / 256):
    POKE P + 8,RB - PEEK (P + 9) * 256: REM RESTDISK
300 POKE P + 10,PA / 256: POKE P + 11,PE / 256:PB = PL / 512:
    REM ANFANG-ENDE-LAENGE
310 IF DB < = PB THEN HOME : PRINT "DISK-DRIVE <= PUFFER?": END
320 HS = INT (DB / PB): POKE P + 12,HS: POKE P + 13,0:
    POKE P + 14,0: REM KOMPLETTDISK-HAUPTSCHUEBE
330 IF DB / PB > HS THEN RS = (DB - HS * PB) * 2: POKE P + 13,1:
    POKE P + 14,RS + PEEK (P + 10): REM KOMPLETTDISK-RESTSCHUB
340 POKE P + 15,0: POKE P + 16,0: POKE P + 17,0: IF RD = 0
    THEN 380: REM KEINE-RESTDISK!
350 IF RB < PB THEN POKE P + 16,1:RS = RB * 2: POKE P + 17,
    RS + PEEK (P + 10): GOTO 380: REM RESTSCHUB-OHNE-HAUPTSCHUB
360 HS = INT (RB / PB): POKE P + 15,HS:
    REM RESTDISK-HAUPTSCHUEBE
370 IF RB / PB > HS THEN RS = (RB - HS * PB) * 2: POKE P + 16,1:
    POKE P + 17,RS + PEEK (P + 10): REM RESTDISK-RESTSCHUB
380 HTAB 11: PRINT "1. MEGA  -> ERPHI": PRINT :
    HTAB 11: PRINT "0. ERPHI -> MEGA": PRINT : PRINT : HTAB 11
390 GET R$: ON R$ < > "1" AND R$ < > "0" GOTO 390:
    POKE P + 18, ASC (R$) - 48: REM RICHTUNG-FLAG
400 IF R$ = "1" THEN VTAB 14: HTAB 10: CALL - 958:
    PRINT "NEUES DATUM TT.MM.JJ"
410 IF R$ = "0" THEN VTAB 14: HTAB 10: CALL - 958:
    PRINT "ALTES DATUM TT.MM.JJ"
420 HTAB 10: INPUT "----------->";D$: IF LEN (D$) < > 8 OR
    MID$ (D$,3,1) < > "." OR MID$ (D$,6,1) < > "." THEN 400
430 FOR X = 1 TO 8: POKE P + 18 + X, ASC ( MID$ (D$,X,1)): NEXT
440 HOME : PRINT KD + RD;" FORMATIERTE DISKETTEN - ETIKETTIERT":
    PRINT : PRINT "#1-";D$;", #2-";D$;", #3-";D$: PRINT :
    PRINT "USW. BEREITLEGEN UND GENAU": PRINT :
    PRINT "IN DERJENIGEN REIHENFOLGE EINLEGEN,": PRINT :
    PRINT "WIE SIE AM BILDSCHIRM ANGEWIESEN WERDEN"
450 PRINT : IF R$ = "1" THEN PRINT "RICHTUNG MEGA -> ERPHI"
460 IF R$ = "0" THEN PRINT "RICHTUNG ERPHI -> MEGA"
470 PRINT : PRINT "OKAY J/N ";: GET X$: ON X$ < > "J" AND
    X$ < > "N" GOTO 440: IF X$ = "N" THEN PRINT :
    PRINT CHR$ (4);"PREFIX";SP$: END
480 PRINT : PRINT CHR$ (4);"PREFIX";SP$:
    PRINT CHR$ (4);"VERIFY";SP$;"PRO.BACKUP.O":
    PRINT CHR$ (4);"BRUN PRO.BACKUP.O": END
```

PRO.BACKUP.O

```
BSAVE PRO.BACKUP.O, A2051, L509

                 1            ORG  $0803
                 2    *
                 3    * PRO.BACKUP.O
                 4    * ============
                 5    *
                 6    * Backup-Programm zwischen
                 7    * Megaboard-Festplatte und
                 8    * Erphi-160-Spur-Subsystem
                 9    * U.Stiehl/Mai/1986
                10    *
                11    RESVKT   EQU  $03F2      ;Vektor
                12    MLI      EQU  $BF00      ;ProDOS
                13    SETVID   EQU  $FD93      ;Video
                14    SETKBD   EQU  $FE89      ;Keyboard
                15    HOME     EQU  $FC58      ;Cls
                16    COUT     EQU  $FDED      ;Print
                17    CROUT    EQU  $FD8E      ;Return
                18    TABV     EQU  $FB5B      ;Vtab
                19    PRBYTE   EQU  $FDDA      ;Hexout
                20    BELL1A   EQU  $FBDD      ;Piepston
                21    RDKEY    EQU  $FD0C      ;Taste
                22    SCRTCH   EQU  $D64B      ;New
                23    OUTSP    EQU  $DB57      ;Space
                24    OUTDO    EQU  $DB5C      ;ORA-Cout
                25    BASIC    EQU  $E003      ;Basic
                26    LINPRT   EQU  $ED24      ;Dezout
                27    *
                28    DISKUNIT EQU  $0300      ;DU+00
                29    HARDUNIT EQU  $0301      ;HU+00
                30    *
                31    * Nicht benutzt:
                32    DISKBLKL EQU  $0302      ;DB-low
                33    DISKBLKH EQU  $0303      ;DB-high
                34    HARDBLKL EQU  $0304      ;HB-low
                35    HARDBLKH EQU  $0305      ;HB-high
                36    *
                37    * Komplettdisks KD + Restdisk RD
                38    *
                39    KOMPDISK EQU  $0306      ;KD>=1
                40    RESTDISK EQU  $0307      ;RD>=0
                41    *
                42    * Nicht benutzt:
                43    RESTBLKL EQU  $0308      ;RB-low
                44    RESTBLKH EQU  $0309      ;RB-high
                45    *
                46    * Pufferanfang PA + Pufferende PE
                47    *
                48    PUFANF   EQU  $030A      ;PA=1000
                49    PUFEND   EQU  $030B      ;PE=9000
                50    *
                51    * KD-Hauptschübe + KD-Restschub
                52    *
                53    KDHS     EQU  $030C      ;HS>=1
                54    KDRS     EQU  $030D      ;RS>=0
                55    KDRSPE   EQU  $030E      ;P-Ende
                56    *
                57    * RD-Hauptschübe + RD-Restschub
                58    *
                59    RDHS     EQU  $030F      ;HS>=0
                60    RDRS     EQU  $0310      ;RS>=0
                61    RDRSPE   EQU  $0311      ;P-Ende
                62    *
                63    * 1: Mega->Erphi  0: Erphi->Mega
                64    *
                65    RICHTUNG EQU  $0312      ;1/0
                66    *
                67    * Datum + Nr. zur Idenfitikation
                68    *
                69    DATUM    EQU  $0313      ;-$031A
                70    DISKNR   EQU  $031B      ;>=0
                71    *
                72    RESET    EQU  $031C      ;-$031E
                73    *
                74    * Temporäre Zähler
                75    *
                76    KDTEMP   EQU  $00FA      ;Kompl.
                77    KDHSTEMP EQU  $00FB      ;Haupt.
                78    RDHSTEMP EQU  $00FC      ;Rest.
                79    PETEMP   EQU  $00FD      ;P-Ende
                80    IND      EQU  $00CE      ;indirekt
                81    *
                82    * Puffer-Verteilung
                83    * -----------------
                84    *
                85    * Optimal 32K-Puffer: $1000-$9000
                86    * d.h.    $8000-Puf.:  4096-36864
                87    * Möglich 36K-Puffer: $0A00-$9A00
                88    * d.h.    $9000-Puf.:  2560-39424
                89    * Maximal 45K-Puffer: $0A00-$BE00
                90    * d.h.    $BE00-Puf.:  2560-48640
                91    *
                92    * Im Maximalfall Bitmap ändern
                93    *
                94    * LDA #$00
                95    * STA $BF6B ;Bitmap
                96    * STA $BF6C
                97    * STA $BF6D
                98    * STA $BF6E
                99    * LDA #%00000011
               100    * STA $BF6F
               101    *
               102    * Reset-Vektor setzen
               103    *
0803: A0 02    104            LDY  #2
0805: B9 F2 03 105    INIT1    LDA  RESVKT,Y
```

```
Ø8Ø8: 99 1C Ø3  1Ø6          STA  RESET,Y
Ø8ØB: 88        1Ø7          DEY
Ø8ØC: 1Ø F7     1Ø8          BPL  INIT1
Ø8ØE: 18        1Ø9  INIT2   CLC
Ø8ØF: 2Ø 4B D6  11Ø          JSR  SCRTCH
Ø812: 2Ø 93 FD  111          JSR  SETVID     ;DOS ab-
Ø815: 2Ø 89 FE  112          JSR  SETKBD     ;hängen
Ø818: A9 27     113          LDA  #<INIT3    ;Reset-
Ø81A: 8D F2 Ø3  114          STA  RESVKT     ;Vektor
Ø81D: A9 Ø8     115          LDA  #>INIT3    ;setzen
Ø81F: 8D F3 Ø3  116          STA  RESVKT+1
Ø822: 49 A5     117          EOR  #$A5       ;Funny
Ø824: 8D F4 Ø3  118          STA  RESVKT+2
                119  *
                12Ø  * Units poken, Richtung beachten
                121  *
Ø827: AC ØØ Ø3  122  INIT3   LDY  DISKUNIT
Ø82A: AE Ø1 Ø3  123          LDX  HARDUNIT
Ø82D: 8C DD Ø8  124          STY  WUNIT      ;Mega->
Ø83Ø: 8E CE Ø8  125          STX  RUNIT      ;Erphi
Ø833: AD 12 Ø3  126          LDA  RICHTUNG   ;Ø/1?
Ø836: DØ Ø6     127          BNE  INIT4      ;<>1
Ø838: 8C CE Ø8  128          STY  RUNIT      ;Erphi
Ø83B: 8E DD Ø8  129          STX  WUNIT      ;->Mega
                13Ø  *
                131  * Mit Hard-Block $ØØØ1 beginnen!
                132  *
Ø83E: AØ Ø1     133  INIT4   LDY  #1         ;$ØØØ1
Ø84Ø: A2 ØØ     134          LDX  #Ø
Ø842: 8C D1 Ø8  135          STY  RBLKL      ;Mega->
Ø845: 8E D2 Ø8  136          STX  RBLKH      ;Erphi
Ø848: AD 12 Ø3  137          LDA  RICHTUNG
Ø84B: DØ Ø6     138          BNE  INIT5
Ø84D: 8C EØ Ø8  139          STY  WBLKL      ;Erphi
Ø85Ø: 8E E1 Ø8  14Ø          STX  WBLKH      ;->Mega
                141  *
                142  * Disk-Nr. + Komplettdisk-Zähler
                143  *
Ø853: A9 ØØ     144  INIT5   LDA  #Ø
Ø855: 8D 1B Ø3  145          STA  DISKNR
Ø858: AD Ø6 Ø3  146          LDA  KOMPDISK
Ø85B: 85 FA     147          STA  KDTEMP
                148  *
                149  * Komplettdisk?
                15Ø  * -------------
                151  *
Ø85D: FØ 2C     152  KOMPDSK? BEQ  RESTDSK?   ;nein(?)
                153  *
                154  * Komplettdisk-Hauptpuffer
                155  *
Ø85F: EE 1B Ø3  156  KDLOOP1 INC  DISKNR
Ø862: C6 FA     157          DEC  KDTEMP
Ø864: AD ØB Ø3  158          LDA  PUFEND
Ø867: 85 FD     159          STA  PETEMP     ;PUFEND
Ø869: 2Ø 34 Ø9  16Ø          JSR  INSERT1
Ø86C: AD ØC Ø3  161          LDA  KDHS       ;Haupt?
Ø86F: FØ Ø9     162          BEQ  KDLOOP3    ;nein(?)
Ø871: 85 FB     163          STA  KDHSTEMP
Ø873: 2Ø ED Ø8  164  KDLOOP2 JSR  READA      ;Haupt
Ø876: C6 FB     165          DEC  KDHSTEMP   ;erneut?
Ø878: DØ F9     166          BNE  KDLOOP2    ;ja
                167  *
                168  * Komplettdisk-Restpuffer
                169  *
Ø87A: AD ØE Ø3  17Ø  KDLOOP3 LDA  KDRSPE     ;Rest
Ø87D: 85 FD     171          STA  PETEMP     ;KDRSPE
Ø87F: AD ØD Ø3  172          LDA  KDRS
Ø882: FØ Ø7     173          BEQ  RESTDSK?   ;nein
Ø884: 2Ø ED Ø8  174          JSR  READA
                175  *
                176  * Noch eine Komplettdisk?
                177  *
Ø887: A5 FA     178          LDA  KDTEMP
Ø889: DØ D4     179          BNE  KDLOOP1
                18Ø  *
                181  * Restdisk?
                182  * ---------
                183  *
Ø88B: AD Ø7 Ø3  184  RESTDSK? LDA  RESTDISK
Ø88E: FØ 26     185          BEQ  EXIT1      ;nein
                186  *
                187  * Restdisk-Hauptpuffer
                188  *
Ø89Ø: EE 1B Ø3  189  RDLOOP1 INC  DISKNR
Ø893: AD ØB Ø3  19Ø          LDA  PUFEND
Ø896: 85 FD     191          STA  PETEMP     ;PUFEND
Ø898: 2Ø 34 Ø9  192          JSR  INSERT1
Ø89B: AD ØF Ø3  193          LDA  RDHS       ;Haupt?
Ø89E: FØ Ø9     194          BEQ  RDLOOP3    ;nein(?)
Ø8AØ: 85 FC     195          STA  RDHSTEMP
Ø8A2: 2Ø ED Ø8  196  RDLOOP2 JSR  READA      ;Haupt
Ø8A5: C6 FC     197          DEC  RDHSTEMP   ;erneut?
Ø8A7: DØ F9     198          BNE  RDLOOP2    ;ja
                199  *
                2ØØ  * Restdisk-Restpuffer
                2Ø1  *
Ø8A9: AD 11 Ø3  2Ø2  RDLOOP3 LDA  RDRSPE     ;Rest
Ø8AC: 85 FD     2Ø3          STA  PETEMP     ;RDRSPE
Ø8AE: AD 1Ø Ø3  2Ø4          LDA  RDRS
Ø8B1: FØ Ø3     2Ø5          BEQ  EXIT1      ;nein
Ø8B3: 2Ø ED Ø8  2Ø6          JSR  READA
                2Ø7  *
Ø8B6: AØ Ø2     2Ø8  EXIT1   LDY  #2
Ø8B8: B9 1C Ø3  2Ø9  EXIT2   LDA  RESET,Y    ;Reset
Ø8BB: 99 F2 Ø3  21Ø          STA  RESVKT,Y   ;wieder
Ø8BE: 88        211          DEY             ;normal
Ø8BF: 1Ø F7     212          BPL  EXIT2
Ø8C1: 4C Ø3 EØ  213          JMP  BASIC      ;Basic
                214  *
                215  *-----------------------------
                216  *
                217  * Diverse Unterroutinen
                218  *
                219  * 1 Block lesen
                22Ø  *
Ø8C4: 2Ø ØØ BF  221  READ    JSR  MLI
Ø8C7: 8Ø        222          HEX  8Ø         ;Read
Ø8C8: CD Ø8     223          DA   RCNT
Ø8CA: BØ 16     224          BCS  MLIERR1
Ø8CC: 6Ø        225          RTS
Ø8CD: Ø3        226  RCNT    HEX  Ø3         ;Count
Ø8CE: ØØ        227  RUNIT   HEX  ØØ         ;gepokt
Ø8CF: ØØ        228  RPUFL   HEX  ØØ         ;stets Ø
Ø8DØ: ØØ        229  RPUFH   HEX  ØØ         ;gepokt
Ø8D1: ØØ        23Ø  RBLKL   HEX  ØØ         ;gepokt
Ø8D2: ØØ        231  RBLKH   HEX  ØØ         ;gepokt
                232  *
                233  * 1 Block schreiben
                234  *
Ø8D3: 2Ø ØØ BF  235  WRITE   JSR  MLI
Ø8D6: 81        236          HEX  81         ;Write
Ø8D7: DC Ø8     237          DA   WCNT
Ø8D9: BØ Ø7     238          BCS  MLIERR1
Ø8DB: 6Ø        239          RTS
Ø8DC: Ø3        24Ø  WCNT    HEX  Ø3         ;Count
Ø8DD: ØØ        241  WUNIT   HEX  ØØ         ;gepokt
Ø8DE: ØØ        242  WPUFL   HEX  ØØ         ;stets Ø
Ø8DF: ØØ        243  WPUFH   HEX  ØØ         ;gepokt
Ø8EØ: ØØ        244  WBLKL   HEX  ØØ         ;gepokt
Ø8E1: ØØ        245  WBLKH   HEX  ØØ         ;gepokt
                246  *
Ø8E2: 48        247  MLIERR1 PHA             ;MLI-
Ø8E3: 2Ø 58 FC  248          JSR  HOME       ;Fehler-
Ø8E6: 68        249          PLA             ;nummer-
Ø8E7: 2Ø DA FD  25Ø          JSR  PRBYTE     ;ausgabe
                251  *
                252  * Numquam retrorsum!
                253  *
Ø8EA: 4C EA Ø8  254  MLIERR2 JMP  MLIERR2     ;Reset
                255  *
                256  * Hauptschübe/Restschub: Read
                257  *
Ø8ED: AD ØA Ø3  258  READA   LDA  PUFANF
Ø8FØ: 8D DØ Ø8  259          STA  RPUFH
Ø8F3: A9 D2     26Ø          LDA  #"R"
Ø8F5: 2Ø ED FD  261          JSR  COUT
Ø8F8: 2Ø C4 Ø8  262  READB   JSR  READ
Ø8FB: EE D1 Ø8  263          INC  RBLKL      ;Blk+1
Ø8FE: DØ Ø3     264          BNE  READC
Ø9ØØ: EE D2 Ø8  265          INC  RBLKH
Ø9Ø3: EE DØ Ø8  266  READC   INC  RPUFH      ;Puf+2
Ø9Ø6: EE DØ Ø8  267          INC  RPUFH
Ø9Ø9: AD DØ Ø8  268          LDA  RPUFH
Ø9ØC: C5 FD     269          CMP  PETEMP
Ø9ØE: 9Ø E8     27Ø          BCC  READB
                271  *
                272  * Hauptschübe/Restschub: Write
                273  *
Ø91Ø: AD ØA Ø3  274  WRITEA  LDA  PUFANF
Ø913: 8D DF Ø8  275          STA  WPUFH
Ø916: A9 D7     276          LDA  #"W"
Ø918: 2Ø ED FD  277          JSR  COUT
Ø91B: 2Ø D3 Ø8  278  WRITEB  JSR  WRITE
Ø91E: EE EØ Ø8  279          INC  WBLKL      ;Blk+1
Ø921: DØ Ø3     28Ø          BNE  WRITEC
Ø923: EE E1 Ø8  281          INC  WBLKH
Ø926: EE DF Ø8  282  WRITEC  INC  WPUFH      ;Puf+2
Ø929: EE DF Ø8  283          INC  WPUFH
Ø92C: AD DF Ø8  284          LDA  WPUFH
Ø92F: C5 FD     285          CMP  PETEMP
Ø931: 9Ø E8     286          BCC  WRITEB
Ø933: 6Ø        287          RTS
                288  *
                289  * Diskette einlegen
```

```
                  290  *
Ø934: 2Ø 58 FC    291  INSERT1  JSR  HOME
Ø937: A9 Ø9       292           LDA  #9          ;Vtab 8
Ø939: 2Ø 5B FB    293           JSR  TABV
Ø93C: AE 1B Ø3    294           LDX  DISKNR      ;low
Ø93F: A9 ØØ       295           LDA  #Ø          ;high=Ø
Ø941: 2Ø 24 ED    296           JSR  LINPRT
Ø944: AØ ØØ       297           LDY  #Ø
Ø946: B9 DD Ø9    298  INSERT2  LDA  EINLEGEN,Y
Ø949: FØ Ø6       299           BEQ  INSERT3
Ø94B: 2Ø ED FD    3ØØ           JSR  COUT
Ø94E: C8          3Ø1           INY
Ø94F: DØ F5       3Ø2           BNE  INSERT2
Ø951: 2Ø ØC FD    3Ø3  INSERT3  JSR  RDKEY
Ø954: C9 D7       3Ø4           CMP  #"W"
Ø956: DØ F9       3Ø5           BNE  INSERT3
Ø958: 2Ø 8E FD    3Ø6           JSR  CROUT
Ø95B: 2Ø 8E FD    3Ø7           JSR  CROUT
                  3Ø8  *
                  3Ø9  * "TT.MM.JJ" + Disknummer
                  31Ø  *
Ø95E: A9 ØØ       311  IDENT    LDA  #Ø
Ø96Ø: 85 CE       312           STA  IND
Ø962: AD ØA Ø3    313           LDA  PUFANF
Ø965: 85 CF       314           STA  IND+1
Ø967: AD 12 Ø3    315           LDA  RICHTUNG
Ø96A: FØ 1F       316           BEQ  IDENTØA
                  317  *
                  318  * auf Block Ø schreiben
                  319  *
Ø96C: A9 ØØ       32Ø  IDENT1A  LDA  #Ø          ;$ØØØØ
Ø96E: 8D EØ Ø8    321           STA  WBLKL       ;Mega->
Ø971: 8D E1 Ø8    322           STA  WBLKH       ;Erphi
Ø974: AD ØA Ø3    323           LDA  PUFANF
Ø977: 8D DF Ø8    324           STA  WPUFH
Ø97A: AØ Ø8       325           LDY  #8
Ø97C: B9 13 Ø3    326  IDENT1B  LDA  DATUM,Y
Ø97F: 91 CE       327           STA  (IND),Y
Ø981: 88          328           DEY
Ø982: 1Ø F8       329           BPL  IDENT1B
Ø984: 2Ø D3 Ø8    33Ø           JSR  WRITE
Ø987: EE EØ Ø8    331           INC  WBLKL       ;$ØØØ1
Ø98A: 6Ø          332           RTS
                  333  *
                  334  * von Block Ø lesen
                  335  *
Ø98B: A9 ØØ       336  IDENTØA  LDA  #Ø          ;$ØØØØ
Ø98D: 8D D1 Ø8    337           STA  RBLKL       ;Erphi
Ø99Ø: 8D D2 Ø8    338           STA  RBLKH       ;->Mega
Ø993: AD ØA Ø3    339           LDA  PUFANF
Ø996: 8D DØ Ø8    34Ø           STA  RPUFH
Ø999: 2Ø C4 Ø8    341           JSR  READ
Ø99C: AØ Ø8       342           LDY  #8
Ø99E: B9 13 Ø3    343  IDENTØB  LDA  DATUM,Y
Ø9A1: D1 CE       344           CMP  (IND),Y
Ø9A3: DØ Ø7       345           BNE  IDENTØC
Ø9A5: 88          346           DEY
Ø9A6: 1Ø F6       347           BPL  IDENTØB
Ø9A8: EE D1 Ø8    348           INC  RBLKL       ;$ØØØ1
Ø9AB: 6Ø          349           RTS
                  35Ø  *
                  351  * Falsche Backup-Diskette
                  352  *
Ø9AC: 2Ø DD FB    353  IDENTØC  JSR  BELL1A
Ø9AF: AØ ØØ       354           LDY  #Ø
Ø9B1: B9 F7 Ø9    355  IDENTØD  LDA  FALSCH,Y
Ø9B4: FØ Ø6       356           BEQ  IDENTØE
Ø9B6: 2Ø ED FD    357           JSR  COUT
Ø9B9: C8          358           INY
Ø9BA: DØ F5       359           BNE  IDENTØD
Ø9BC: AØ Ø8       36Ø  IDENTØE  LDY  #8
Ø9BE: B1 CE       361           LDA  (IND),Y     ;Nr.
Ø9CØ: AA          362           TAX              ;low
Ø9C1: A9 ØØ       363           LDA  #Ø          ;high=Ø
Ø9C3: 2Ø 24 ED    364           JSR  LINPRT
Ø9C6: A9 AD       365           LDA  #"-"
Ø9C8: AØ ØØ       366           LDY  #Ø
Ø9CA: 2Ø 5C DB    367  IDENTØF  JSR  OUTDO
Ø9CD: B1 CE       368           LDA  (IND),Y
Ø9CF: C8          369           INY
Ø9DØ: CØ Ø9       37Ø           CPY  #9
Ø9D2: DØ F6       371           BNE  IDENTØF
Ø9D4: 2Ø 57 DB    372           JSR  OUTSP
Ø9D7: 2Ø ØC FD    373           JSR  RDKEY
Ø9DA: 4C 34 Ø9    374           JMP  INSERT1
                  375  *
Ø9DD: AE AØ C4    376  EINLEGEN ASC  ". DISK"
Ø9E3: AØ C5 C9    377           ASC  " EINLEGEN"
Ø9EC: 8D 8D       378           HEX  8D8D
Ø9EE: D7 A8 C5    379           ASC  "W(EITER "
Ø9F6: ØØ          38Ø           HEX  ØØ
Ø9F7: C6 C1 CC    381  FALSCH   ASC  "FALSCH! "
Ø9FF: ØØ          382           HEX  ØØ
```

KOPY.160.SPUR

```
1ØØ REM *KOPY.16Ø.SPUR*
11Ø DU = 6:HU = 6
12Ø DB = 128Ø:HB = 128Ø
13Ø PA = Ø256Ø: REM $ØAØØ
14Ø PE = 4864Ø: REM $BEØØ
15Ø PL = PE - PA: IF PL / 512 < > INT (PL / 512) THEN
    PRINT "NUR GANZE BLOCKANZAHL!": END
16Ø P = 768: REM PAGE-3-PARAMETER
17Ø POKE P,DU * 16 + Ø: REM +128-FUER-D2
18Ø POKE P + 1,HU * 16 + 128: REM +128-FUER-D2
19Ø POKE P + 3, INT (DB / 256): POKE P + 2,DB -
    PEEK (P + 3) * 256: REM DISK-BLOCKANZAHL
2ØØ POKE P + 5, INT (HB / 256): POKE P + 4,HB -
    PEEK (P + 5) * 256: REM HARD-BLOCKANZAHL
21Ø KD = 1: POKE P + 6,KD:RD = Ø: POKE P + 7,RD: POKE P + 8,Ø:
    POKE P + 9,Ø: REM KOMPLETTDISK
22Ø POKE P + 1Ø,PA / 256: POKE P + 11,PE / 256:PB = PL / 512:
    REM ANFANG-ENDE-LAENGE
23Ø HS = INT (DB / PB): POKE P + 12,HS: POKE P + 13,Ø:
    POKE P + 14,Ø: REM KOMPLETTDISK-HAUPTSCHUEBE
24Ø IF DB / PB > HS THEN RS = (DB - HS * PB) * 2: POKE P + 13,1:
     POKE P + 14,RS + PEEK (P + 1Ø): REM KOMPLETTDISK-RESTSCHUB
25Ø POKE P + 15,Ø: POKE P + 16,Ø: POKE P + 17,Ø:
    REM KEINE-RESTDISK!
26Ø PRINT CHR$ (21): TEXT : HOME : PRINT : INVERSE :
    HTAB 1Ø: PRINT "** 16Ø-SPUR-KOPY **":
    HTAB 1Ø: PRINT "** VON U. STIEHL **": NORMAL :
    PRINT CHR$ (4);"PREFIX": INPUT SP$: REM START-PREFIX
27Ø HTAB 1Ø: PRINT "ORIGINAL IN DRIVE 1": PRINT :
    HTAB 1Ø: PRINT "DUPLIKAT IN DRIVE 2"
28Ø PRINT : PRINT CHR$ (4);"BRUN";SP$;"KOPY.16Ø.SPUR.O": END
```

KOPY.160.SPUR.O

BSAVE KOPY.16Ø.SPUR.O, A2Ø51,L474
(Erst linke, dann rechte Spalte eingeben)

```
$Ø8ØØ: ØØ ØØ ØØ A9 ØØ 8D 12 Ø3  $Ø9ØØ: AØ CA AF CE AØ ØØ 4C Ø3
$Ø8Ø8: 8D 6B BF 8D 6C BF 8D 6D  $Ø9Ø8: EØ 2Ø ØØ BF 8Ø 12 Ø9 BØ
$Ø81Ø: BF 8D 6E BF A9 Ø3 8D 6F  $Ø91Ø: 16 6Ø Ø3 ØØ ØØ ØØ ØØ ØØ
$Ø818: BF AØ Ø2 B9 F2 Ø3 99 1C  $Ø918: 2Ø ØØ BF 81 21 Ø9 BØ Ø7
$Ø82Ø: Ø3 88 1Ø F7 18 2Ø 4B D6  $Ø92Ø: 6Ø Ø3 ØØ ØØ ØØ ØØ ØØ 48
$Ø828: 2Ø 93 FD 2Ø 89 FE A9 3D  $Ø928: 2Ø 58 FC 68 2Ø DA FD 4C
$Ø83Ø: 8D F2 Ø3 A9 Ø8 8D F3 Ø3  $Ø93Ø: 2F Ø9 AD ØA Ø3 8D 15 Ø9
$Ø838: 49 A5 8D F4 Ø3 AC ØØ Ø3  $Ø938: A9 D2 2Ø ED FD 2Ø Ø9 Ø9
$Ø84Ø: AE Ø1 Ø3 8C 22 Ø9 8E 13  $Ø94Ø: EE 16 Ø9 DØ Ø3 EE 17 Ø9
$Ø848: Ø9 AD 12 Ø3 DØ Ø6 8C 13  $Ø948: EE 15 Ø9 EE 15 Ø9 AD 15
$Ø85Ø: Ø9 8E 22 Ø9 AØ ØØ A2 ØØ  $Ø95Ø: Ø9 C5 FD 9Ø E8 AD ØA Ø3
$Ø858: 8C 16 Ø9 8E 17 Ø9 AD 12  $Ø958: 8D 24 Ø9 A9 D7 2Ø ED FD
$Ø86Ø: Ø3 DØ Ø6 8C 25 Ø9 8E 26  $Ø96Ø: 2Ø 18 Ø9 EE 25 Ø9 DØ Ø3
$Ø868: Ø9 A9 ØØ 8D 1B Ø3 AD Ø6  $Ø968: EE 26 Ø9 EE 24 Ø9 EE 24
$Ø87Ø: Ø3 85 FA FØ 2C EE 1B Ø3  $Ø97Ø: Ø9 AD 24 Ø9 C5 FD 9Ø E8
$Ø878: C6 FA AD ØB Ø3 85 FD 2Ø  $Ø978: 6Ø 2Ø 58 FC A9 Ø5 2Ø 5B
$Ø88Ø: 79 Ø9 AD ØC Ø3 FØ Ø9 85  $Ø98Ø: FB AØ ØØ B9 AA Ø9 FØ Ø6
$Ø888: FB 2Ø 32 Ø9 C6 FB DØ F9  $Ø988: 2Ø ED FD C8 DØ F5 2Ø ØC
$Ø89Ø: AD ØE Ø3 85 FD AD ØD Ø3  $Ø99Ø: FD C9 D7 DØ F9 2Ø 8E FD
$Ø898: FØ Ø7 2Ø 32 Ø9 A5 FA DØ  $Ø998: 2Ø 8E FD A9 ØØ 8D 16 Ø9
$Ø8AØ: D4 AD Ø7 Ø3 FØ 26 EE 1B  $Ø9AØ: 8D 17 Ø9 AD ØA Ø3 8D 15
$Ø8A8: Ø3 AD ØB Ø3 85 FD 2Ø 79  $Ø9A8: Ø9 6Ø CF D2 C9 C7 C9 CE
$Ø8BØ: Ø9 AD ØF Ø3 FØ Ø9 85 FC  $Ø9BØ: C1 CC AØ C9 CE AØ C4 D2
$Ø8B8: 2Ø 32 Ø9 C6 FC DØ F9 AD  $Ø9B8: C9 D6 C5 AØ B1 8D 8D C4
$Ø8CØ: 11 Ø3 85 FD AD 1Ø Ø3 FØ  $Ø9CØ: D5 DØ CC C9 CB C1 D4 AØ
$Ø8C8: Ø3 2Ø 32 Ø9 AØ Ø2 B9 1C  $Ø9C8: C9 CE AØ C4 D2 C9 D6 C5
$Ø8DØ: Ø3 99 F2 Ø3 88 1Ø F7 2Ø  $Ø9DØ: AØ B2 8D 8D D7 A8 C5 C9
$Ø8D8: 58 FC A9 Ø5 2Ø 5B FB AØ  $Ø9D8: D4 C5 D2 AØ ØØ ØØ ØØ ØØ
$Ø8EØ: ØØ B9 FA Ø8 FØ Ø6 2Ø ED
$Ø8E8: FD C8 DØ F5 2Ø ØC FD C9
$Ø8FØ: CE FØ 13 C9 CA DØ F5 4C
$Ø8F8: Ø3 Ø8 C5 D2 CE C5 D5 D4
```

SUPERDUMP für Apple-Pascal 1.1 und 1.2

Grafik-Ausdruck für Epson-FX80 und Imagewriter I

von Jürgen Geiß

Auf Wunsch sehr vieler Peeker-Leser, die Grafiken mittels SUPERDUMP auch in Pascal ausgeben lassen wollen, soll hier die Version für Apple-Pascal 1.1 bzw. 1.2 gebracht werden. Da der Assemblerteil fast der gleiche wie bei der in Peeker, Heft 6/85 vorgestellten DOS-3.3-Version ist, kann er wegen der Länge nicht abgedruckt werden. Er befindet sich aber auf der Peeker-Sammeldisk.

Das Drucken mittels SUPERDUMP in Pascal kann wesentlich eleganter als in Basic vollzogen werden. Es gibt hier keine Peeks und Pokes, sondern vordefinierte Datentypen, die Variablen enthalten, deren Wertebereich anzeigt, welche Werte erlaubt sind. Diese Datentypen werden im Interfaceteil der **Unit SuperDump** deklariert. Dort befinden sich auch die Deklarationen von acht Prozeduren, die die Handhabung der Grafik erleichtern sollten (s. **Listing 1: Unit SuperDump**). Doch zuerst zu den Typen:

1. Control-Block

Der **CtrlBlk** entspricht genau dem aus Peeker, Heft 6/85. In ihm befinden sich die druckerspezifischen Steuerzeichen, die aus 4 Strings zu je 15 Zeichen gebildet werden.

PINIT ist der String, der einmal zu Beginn des Druckvorgangs gesendet wird. Darin können beispielsweise die Steuercodes für den Zeilenabstand untergebracht werden.

PLINE ist ein String, der vor jeder auszugebenden Druckzeile gesendet wird. Dort müssen sich die Steuerzeichen zum Umschalten des Druckers auf den Grafikbetrieb befinden.

In **PLF1** müssen sich die Steuerzeichen befinden, die das Papier um 1/216" oder 1/144" (Zoll) vorwärts transportieren (wird nur bei 2- und 4-fach-Druck benutzt).

In **PEXIT** schließlich müssen sich die Steuerzeichen befinden, die zum Abschluß eines Druckvorgangs gesendet werden sollen.

In **Listing 2: Print, Epson-Version** befindet sich ein Beispiel einer Vorbelegung des CtrlBlocks für den Epson-Drucker FX-80, in **Listing 3: Print, Imagewriter** entsprechendes für den Imagewriter I. (Anm. d.Red.: Damit nicht wieder eine Flut von Zuschriften auf die Redaktion niederprasselt, sei mit Nachdruck darauf hingewiesen, daß SUPERDUMP für Pascal auf einem Apple IIe (a) mit Epson FX-80 und Parallelschnittstelle und (b) mit Imagewriter I und Super-Serial-Card getestet wurde. Die Anpassung an andere Drucker, z.B. von Star, Okidata usw., kann mangels Hardware von der Redaktion nicht vorgenommen werden. Wenden Sie sich direkt an den Lieferanten oder den jeweiligen Drucker-Hersteller.)

2. Parameter

Um nicht eine Vielzahl von Parametern an die Dump-Prozedur, die das Bild druckt, übergeben zu müssen, wurden diese im Record Dumptype zusammengefaßt. Dieser enthält im einzelnen:

CtrlAdr: Darin muß sich die Adresse einer Variablen vom Typ CtrlBlk befinden. Um die Adressen einer Variablen zu erhalten, kann man sich der Funktion Adr bedienen.

Impression: Druckschwärze (0 = normal, 1 = Doppeldruck, 2 = Vierfachdruck).

XOR: 0 bedeutet Bild schwarz auf weiß ausgeben, 255 weiß auf schwarz.

Direction: 0 = Horizontaldruck, 1 = Vertikaldruck.

Frame: 0 = kein Rahmen, 1 = Rahmen um das Bild zeichnen.

PicAddr: Bildadresse, Seite 1 = 8192, Seite 2 = 16384

Rfactor: Anzahl der Wiederholungen des gleichen Bildes nebeneinander

Rdistance: Anzahl der Bildpunkte, die zwischen 2 oder mehreren aufeinanderfolgenden Bildern frei bleiben soll.

Left, Right, Top und Bottom: Linker, rechter, oberer und unterer Bildrand.

Xfactor, Yfactor: Dehnung des Bildes in X- bzw. Y-Richtung.

Hires560: 0 = normale 280 * 192 Grafik, 1 = 560 * 192 (Double Hires) Grafik.

AutoLF: 0 = sende LF nach CR, 1 = sende kein LF nach CR.

DiffPics: Anzahl der verschiedenen Teilbilder (für große Bilder).

Needle: Drucknadel-Richtung, 0 = Epson-Drucker, 1 = Imagewriter

Comptype: 0 = Apple II+ oder Apple IIe mit parallelem Interface, 1 = Apple II+ oder Apple IIe mit Super-Serial-Card, 2 = Apple IIc.

3. Prozeduren der Unit

Um ein Bild zu drucken, muß die Prozedur Dump mit einer Variablen des eben erläuterten Typs aufgerufen werden (s. Listing 2).

Die Prozedur **MoveAuxMain** verschiebt den Speicher aus dem Auxiliary-RAM ab Adresse $2000 des Apple IIe oder IIc in das Main-RAM nach Adresse $4000.

Die Prozedur **MoveMainAux** verschiebt von Main-RAM $2000 nach Aux-RAM $4000.

Mit diesen beiden Prozeduren können beispielsweise Double-Hires-Bilder erstellt, in das Main-RAM geschoben und anschließend abgespeichert werden.

Convert560 konvertiert ein Double-Hires-Bild in 2 normale Hires-Bilder mit linker Hälfte in Seite 1 und rechter Hälfte in Seite 2.

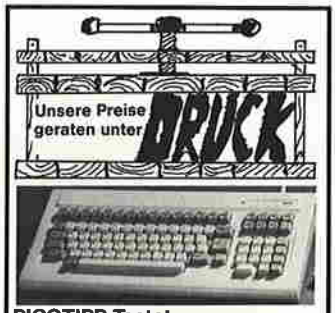
Convert280 konvertiert zwei normale Hires-Bilder ($2000 - $5FFF) in ein Double-Hires-Bild.
Da oft ein Bild nur normal ausgegeben werden soll, d.h. ohne Verzerrung oder sonstigen Besonderheiten, wird die Prozedur **SetDefault** verwendet. Sie stellt alle Variablen des Dump-Records auf Standardwerte, so daß anschließend bei Bedarf nur noch wenige Änderungen vorgenommen werden müssen.
Um die Anzahl der Bildpunkte zu ermitteln, die in jeder Zeile an den Drucker gesendet werden, muß die Prozedur **GetGrafBytes** benutzt werden. Eingabewert ist eine Dumptype-Variable, Ausgabewerte sind eine Integerzahl sowie ein String, in denen sich dann die Anzahl der Bildpunkte befinden. Für einen Epson-Drucker wird die Integerzahl, für einen Imagewriter der String interessant sein. Bei einem Bild von 280 * 192 befindet sich in der Variablen Bytes die Integerzahl 280, im Bytestr der String „0280". Im Beispiel in Listing 2 erkennt man die Benutzung von GetGrafBytes.
Die letzte Prozedur in SuperDump ist die Funktion **Adr**. Sie liefert die Adresse einer beliebigen Variablen im Speicher. Die Funktion wird benutzt, um die Adresse des Kontrollblockes zu ermitteln, um sie der Varibalen DumpParams.CtrlAdr zuzuweisen.

4. Implementation-Teil

Im Implementation-Teil wird die Implementierung der einzelnen Prozeduren aufgezeigt.
Um nun ein Bild drucken lassen zu können, muß nach dem Programmkopf ein
uses TurtleGraphics, SuperDump;
stehen, um dem Programm die neuen Typen und Prozeduren mitzuteilen.
Im Beispielprogramm Print werden 4 Variablen benötigt:
CtrlBlock: CtrlBlk ist der Kontrollblock mit den Steuerzeichen.
DumpParams: Dumptype enthält sämtliche zu übergebenden Parameter.
S: string und *Bytes: integer* werden für GetGrafBytes benötigt.
Das Programm initialisiert die Grafik und zeichnet ein Dreieck mit den Koordinaten (0, 0), (279, 191) und (0, 191). Anschließend wird SetDefault benutzt, um nicht alle Parameter selbst einstellen zu müssen. Die Adresse des Kontrollblockes wird ermittelt und der Variablen DumpParams.CtrlAdr zugewiesen. Nun muß noch die Anzahl der Grafikpunkte durch einen Aufruf von GetGrafBytes ermittelt werden. Bevor gedruckt werden soll, muß der Kontrollblock vorbesetzt werden.
In unserem Beispiel in Listing 2 handelt es sich um einen Epson-Drucker. Die Strings Pinit, Pline, Plf1 und Pexit haben eine Länge von 3, 4, 3 bzw. 2 Zeichen und werden deshalb mit entsprechend vielen Zeichen gefüllt („-"), damit die Länge der Strings definiert ist. Dann werden die Steuerzeichen eingesetzt. Am Ende genügt ein Aufruf von Dump (DumpParams), um das Bild an den Drucker zu schicken.
Für den Imagewriter muß lediglich der Kontrollblock geändert und der Computertyp eingetellt werden (s. Peeker 6/85). Die Ausgabe auf den Drucker erfolgt im Assemblerteil der Unit SuperDump nicht durch Anspringen der Pascal-I/O-Tabelle, d.h. daß keine Steuerzeichen für das Interface gesendet werden dürfen (wie z.B. Ctrl-I, Z), da diese sonst an den Drucker weitergeleitet werden würden. Die Initialisierung des Kontrollblockes für Imagewriter könnte dann also aussehen wie in Listing 3.

Kurzhinweise

1. Zweck:
Ausgabe von Hires- und Double-Hires-Bildern auf den Epson FX-80 oder Imagewriter I.
2. Konfiguration:
Apple II+/e, Apple Pascal 1.1 oder 1.2; FX-80 oder Imagewriter I. Achtung: IIc und Imagewriter II wurden nicht von der Redaktion getestet!
3. Test:
Exec EPSON oder IMAGEWRITER nach (!) der Einbindung in Library (siehe 5)
4. Sammeldisk:
Die Sammeldisk # 18 ist eine kombinierte Pascal-DOS-Diskette und enthält in Pascal-Directory folgende Dateien:
CALLDUMP.TEXT (40 Blöcke)
CALLDUMP.CODE (26 Blöcke)
SUPERDUMP.TEXT (14 Blöcke)
SUPERDUMP.LIB (10 Blöcke)
EPSON.TEXT (4 Blöcke)
EPSON.CODE (2 Blöcke)
IMAGEWRITR.TEXT (4 Blöcke)
IMAGEWRITR.CODE (2 Blöcke)
LIB.TEXT (4 Blöcke)
FUELLER (168 Blöcke; leer!)
5. Einbindung von SUPERDUMP.LIB in SYSTEM.LIBRARY:
a) Starten von LIBRARY.CODE auf der APPLE3: Diskette
b) Auf die Frage Output Code File mit SYSTEM.LIBRARY antworten
c) Auf die Frage Link Code File mit SYSTEM.LIBRARY antworten
d) Alle Slots mit dem "=" Befehl kopieren
e) Mittels New „SUPERDUMP.LIB" holen
f) Slot 1 nach Slot 7 oder anderen freien Slot kopieren (1 <LEERTASTE> 7)
g) Programm mit „q" und anschließendem <RETURN> verlassen
h) Testprogramm EPSON.CODE oder IMAGEWRITR.CODE kann jetzt ausgeführt werden.



PICOTIPP Tastatur
deutsch, für Apple* II+ und Kompatible
– 95 Tasten, Deutsche Tastatur nach DIN
– 12 Funktionstasten über EPROM 4-fach belegbar
– 7 Funktionstasten über Tastatur frei definierbar
– separater Cursorblock
– Cursorsteuerung unter DOS und CP/M aktiv
– Caps Lock
– Neigungswinkel einstellbar
– komplett mit Seriell-parallel-wandelbarem Kabel direkt austauschbar mit Standard-Tastatur

Interfaces/Tastaturen **DM 198.–**

MUSICAL-CARD	DM 99,00
CLOCK-CARD (Time II)	DM 99,00
RS 232 INTERFACE	DM 89,00
DISK-CONTROLLER	DM 69,00
16 kRAM CARD	DM 50,00
LANGUAGE-CARD	DM 60,00
80 ZEICHEN 64 kB RAM nur IIe	DM 79,00
COMMUNIKATION CARD	DM 99,00
SUPER-PARALLEL-CARD	DM 200,00
PARALLEL-CARD	DM 100,00
RGB-CARD	DM 99,00
8088 CARD	DM 300,00
MOTHERBOARD 64 kB RAM 6502/Z80	DM 280,00
MOTHERBOARD 48 kB RAM	DM 250,00
TASTATUR MAK II	DM 168,00
PICOTIPP TASTATUR deutsch	DM 198,00

LIEFERUNG NUR SOLANGE VORRAT REICHT!

*Apple ist ein eingetragenes Warenzeichen der Apple Computer Inc.

picotron gmbh
sigmaringer straße 34
d-7798 pfullendorf
telefon 07552/399

Die ProDOS-Analyse

Version 1.0.1, 1.0.2, 1.1.1

von Arne Schäpers

1985, 470 S., kart., DM 68,–
ISBN 3-7785-1134-3

„Die ProDOS Analyse" ist die umfangreichste und detaillierteste Darstellung, die jemals ein Apple-Betriebssystem erfahren hat. Wer die „Innereien" von ProDOS bis zum letzten Byte, z. T. bis ins letzte Bit kennenlernen möchte, braucht dieses Buch. Das komplette Betriebssystem (Urlader, MLI, Disk-Driver, RAM-Disk-Driver und Uhr-Routine) mit Ausnahme des BASIC-SYSTEM wird mit umfangreichen Kommentaren und Übersichtstabellen disassembliert. Dabei werden alle bisherigen Versionen von 1.0.1 bis 1.1.1 berücksichtigt. „Die ProDOS-Analyse" beschreibt erstmals auch mehrere Programmierfehler, die bis in die neueste Version zu finden sind. Auch die nicht im „Technical Reference Manual" aufgeführten Eigenschaften von ProDOS werden analysiert und beschrieben, z. B. die vertrackten eingebauten Testroutinen zur Identifikation der verschiedenen Apple-II-Modelle und eventueller Nachbaugeräte. Programmierer, die ProDOS versionsabhängig „patchen" möchten, erhalten hier den genauen Überblick, wo was geändert werden muß, damit dies keine negativen Konsequenzen hat. Durch die minutiöse theoretische Sezierung von ProDOS eröffnen sich völlig neue programmierpraktische Perspektiven.

Dr. Alfred Hüthig Verlag · 6900 Heidelberg · Postfach 10 28 69

{Listing 1: Unit SuperDump}

```
unit SuperDump;
intrinsic code 26;
interface

type CtrlStr  = string [15];
     CtrlBlk  = record
                  Pinit : CtrlStr;
                  Pline : CtrlStr;
                  Plfl  : CtrlStr;
                  Pexit : CtrlStr
                end; {CtrlBlk}
     Dumptype = record
                  CtrlAdr    : integer;
                  Impression : Ø..2;
                  XOR        : Ø..255;
                  Direction  : Ø..1;
                  Frame      : Ø..1;
                  PicAddr    : integer;
                  Rfactor    : integer;
                  Rdistance  : integer;
                  Left       : Ø..559;
                  Right      : Ø..559;
                  Top        : Ø..191;
                  Bottom     : Ø..191;
                  Xfactor    : 1..31;
                  Yfactor    : 1..31;
                  Hires56Ø   : Ø..1;
                  AutoLF     : Ø..1;
                  DiffPics   : 1..8;
                  Needle     : Ø..1;
                  Comptype   : Ø..2
                end; {Dumptype}

procedure Dump (DumpParams : Dumptype);
procedure MoveAuxMain;
procedure MoveMainAux;
procedure Convert56Ø;
procedure Convert28Ø;
procedure SetDefault (var DumpParams : Dumptype);
procedure GetGrafBytes (DumpParams  : Dumptype;
                        var Bytes   : integer;
                        var Bytestr : string);
function  Adr (var AnyVariable) : integer;

{-----------------------------------------------------------------}

implementation
{$C unit SuperDump von Juergen Geiss}
{$I-}
{$R-}
```

```
type Paramtype   = record
                     CtrlBlk    : Ø..1Ø24;
                     CtrlAdr    : integer;
                     Impression : Ø..2;
                     XOR        : Ø..255;
                     Direction  : Ø..1;
                     Frame      : Ø..1;
                     PicAddr    : integer;
                     Rfactor    : integer;
                     Rdistance  : integer;
                     Left       : integer;
                     Right      : integer;
                     Top        : Ø..191;
                     Bottom     : Ø..191;
                     Xfactor    : 1..31;
                     Yfactor    : 1..31;
                     Hires56Ø   : Ø..1;
                     AutoLF     : Ø..1;
                     DiffPics   : 1..8;
                     Needle     : Ø..1;
                     Comptype   : Ø..2;
                     Command    : Ø..4
                       {Ø = dump
                        1 = Aux  $2ØØØ -> Main $4ØØØ
                        2 = Main $4ØØØ -> Aux  $2ØØØ
                        3 = convert   56Ø -> 2*28Ø
                        4 = convert 2*28Ø ->   56Ø}
                   end; {Paramtype}

{-----------------------------------------------------------------}

procedure CallDump (Params : Paramtype); external;

{-----------------------------------------------------------------}

procedure Dump {DumpParams : Dumptype};

var Params : Paramtype;

begin {Dump}
  with Params do
  begin
    CtrlBlk    := Ø;
    CtrlAdr    := DumpParams.CtrlAdr;
    Impression := DumpParams.Impression;
    XOR        := DumpParams.XOR;
    Direction  := DumpParams.Direction;
    Frame      := DumpParams.Frame;
    PicAddr    := DumpParams.PicAddr;
    Rfactor    := DumpParams.Rfactor;
    Rdistance  := DumpParams.Rdistance;
    Left       := DumpParams.Left mod 7 +
                  DumpParams.Left div 7 * 256;
    Right      := DumpParams.Right mod 7 +
                  DumpParams.Right div 7 * 256;
    Top        := DumpParams.Top;
    Bottom     := DumpParams.Bottom;
    Xfactor    := DumpParams.Xfactor;
    Yfactor    := DumpParams.Yfactor;
    Hires56Ø   := DumpParams.Hires56Ø;
    AutoLF     := DumpParams.AutoLF;
    DiffPics   := DumpParams.DiffPics;
    Needle     := DumpParams.Needle;
    Comptype   := DumpParams.Comptype;
    Command    := Ø
  end; {with}

  CallDump (Params)
end; {MoveAuxMain}

{-----------------------------------------------------------------}

procedure MoveAuxMain;

var Params : Paramtype;

begin {MoveAuxMain}
  Params.Command := 1;
  CallDump (Params)
end; {MoveAuxMain}

{-----------------------------------------------------------------}

procedure MoveMainAux;

var Params : Paramtype;

begin {MoveMainAux}
  Params.Command := 2;
  CallDump (Params)
end; {MoveMainAux}
```

Achtung Autoren!

Wenn Sie ein Buch oder eine Software nicht bei uns finden, warum schreiben Sie diese nicht für uns?
Zur Erweiterung unseres Verlagsprogrammes in den Sparten EDV und Elektrotechnik suchen wir ständig Autoren, die wie wir »Qualität« und »Zuverlässigkeit« hoch einschätzen. Im Bereich Home- und Personal-Computer haben auch schreibfreudige Hobby-EDVler die Möglichkeit, ihre besonderen Kenntnisse umzusetzen. Die Erfahrung eines renommierten Fachbuch-Verlages wird Ihnen schon beim Abfassen des Manuskriptes zu Nutze kommen. Und unsere Konditionen sind interessant.

Sprechen Sie uns doch einmal ganz unverbindlich an:

Hüthig

Dr. Alfred Hüthig Verlag
Abt. Buchverlag
Postfach 10 28 69
6900 Heidelberg

Für Apple II, IIe

Z-80-Karte	69,—	80-Zeichen-Karte m. Softswitch, nur f. II Plus kombat.	149,—
Disk-Interface	69,—	Speech-Karte	55,—
Centronics-Interf. m. Kabel	79,—	Clock-Karte	129,—
16-K-RAM-Karte	79,—	Komp 2E Apple 2E kompatible, Rechner 64K im 2E-Design, ohne Firmware	797,—
RS-232-Karte	119,—	80Z + 64K-Karte für 2E kompatible	99,—
Eprommer (4, 8, 16 K)	139,—	Motherboard 2E 2E kompatible, ohne Firmware	488,—
128-K-RAM-Karte	249,—		
256-KB-RAM-Karte	399,—		
Motherboard 48K II Plus kombat. 6502, 48 K o. Firmware	399,—		

Händleranfragen erwünscht
Apple-Info 1,— DM (Porto)

Klaus Jeschke
Hard-, Software
Viertstr. 3—13
6233 Kelkheim
☎ (06198) 9069

WARGAMES

Militärische Konfliktsimulationen mit bis zu 100 (!) Stunden Spieldauer. Die Herausforderung an jeden Strategiespieler. Farbinfo für APPLE gegen DM 0,80 Rückporto.

SSI

THOMAS MÜLLER
COMPUTER—SERVICE
Postfach 2526 7600 Offenburg

ProDOS-Editor
Die ursprünglich geplante zusätzliche Version für 80-Zeichendarstellung wird nicht mehr erscheinen. Der ProDOS-Editor wird daher nach wie vor in der 40-Z/Z-Version ausgeliefert.

Softbreaker als Quellcode
Aus Gründen der geänderten Hardware-Interrupt- und Resetroutinen funktionierte der Softbreaker, wie auch in der Anleitung angegeben, nur auf dem ursprünglichen IIe mit den alten ROMs und somit nicht auf dem IIc (mit alten und neuen ROMs) und dem IIe mit neuen ROMs. Da jetzt vermehrt die neuen IIe-ROMs eingesetzt werden, werden ab sofort die Manuals des Softbreakers zum Preis von DM 20,– (ursprünglich DM 48,–) mit einer neuen Diskette vertrieben, die 2 Objektcode-Varianten sowie zusätzlich den kompletten Quellcode enthält, da man die neuen IIe-ROMs oft überlisten kann, indem man absolute Einsprünge ($C600 usw.) benutzt. Wer nach dem 1.1.86 den Softbreaker regulär bestellt hat, erhält die neue Diskette kostenlos nachgeliefert, wenn die kostenlose Nachbestellung bis zum 30.6.86 beim Hüthig Software Service eingeht.

Applesoft-Editor mit Merge-Befehl
Der als Treue-Prämie für Fortsetzungsbezieher erhältliche Applesoft-Editor namens MACROEDITOR auf der Sammeldisk #16 hat einen zusätzlichen Befehl, der in der Anleitung in Peeker, 4/86, S. 13 nicht erwähnt wurde. Mit
NEW
LOAD HAUPTPROGRAMM
&R1,1 (nur bei Zeilenkonflikten)
&P
kann man das Hauptprogramm in einen Schutzbereich übernehmen, wonach man mit
LOAD UNTERPROGRAMM
&R1000,1 (nur bei Zeilenkonflikten)
&M
Unterprogramm und Hauptprogramm vereinen („mergen") kann.

Wer extrem überlange Zeilen editieren muß, kann den LIST-Befehl vorübergehend patchen (LC erforderlich!):
ü CALL -151
* C081 C081
* D000 <D000.FFFFM
* C083 C083
* D748:0C (statt 07; Space vor Token)
* D762: A4 85 D0 AC (statt A9 20 D0 98; Space nach Token)
* D70B: 4C 12 D7 (statt 20 FB DA; Poke-33,33-Patch)
* Ctrl-C
ü K + Zeilennummer

Hüthig Software Service · Postfach 10 28 69 · 6900 Heidelberg 1

```
{-----------------------------------------------------------}

procedure Convert56Ø;

var Params : Paramtype;

begin {Convert56Ø}
  Params.Command := 3;
  CallDump (Params)
end; {Convert56Ø}

{-----------------------------------------------------------}

procedure Convert28Ø;

var Params : Paramtype;

begin {Convert28Ø}
  Params.Command := 4;
  CallDump (Params)
end; {Convert28Ø}

{-----------------------------------------------------------}

procedure SetDefault {var DumpParams : Dumptype};

begin {SetDefault}
  fillchar (DumpParams, size_of (DumpParams), Ø);

  with DumpParams do
  begin
    PicAddr   := 8192;
    Rfactor   :=    1;
    Right     :=  279;
    Bottom    :=  191;
    Xfactor   :=    1;
    Yfactor   :=    1;
    DiffPics  :=    1
  end {with}
end; {SetDefault}

{-----------------------------------------------------------}

procedure GetGrafBytes {DumpParams  : Dumptype;
                        var Bytes   : integer;
                        var Bytestr : string};
var B : integer;
    I : integer;

begin {GetGrafBytes}
  with DumpParams do
  begin
    if Direction = Ø
    then B := Xfactor * (Right - Left + 1)
    else B := Yfactor * (Bottom - Top + 1);

    B       := (B + Rdistance) * Rfactor * DiffPics;
    B       := abs (B) mod 1ØØØØ;
    Bytes   := B;
    Bytestr := 'ØØØØ';

    for I := 4 downto 1 do
    begin
      Bytestr [I] := chr (ord ('Ø') + B mod 1Ø);
      B := B div 1Ø
    end {while}
  end {with}
end; {GetGrafBytes}

{-----------------------------------------------------------}

function Adr {(var AnyVariable) : integer}; external;

{-----------------------------------------------------------}

begin {SuperDump}
  {keine Initialisierung notwendig}
end {SuperDump}.
```

{Listing 2: Program Print, Epson-Version}

```
program Print;
uses TurtleGraphics, SuperDump;

const ESC = 27;

var   CtrlBlock  : CtrlBlk;
      DumpParams : Dumptype;
      S          : string;
      Bytes      : integer;

begin {Print}
  Initturtle;
  moveto (Ø, Ø);
  pencolor (white);
  moveto (279, 191);
  moveto (Ø, 191);
  moveto (Ø, Ø);

  SetDefault (DumpParams);
  DumpParams.CtrlAdr := Adr (CtrlBlock);
  GetGrafBytes (DumpParams, Bytes, S);

  with CtrlBlock do {EPSON}
  begin
    Pinit := '-3-';
    Pline := '-K--';
    Plfl  := '-J-';
    Pexit := '-2';

    Pinit [1] := chr (ESC);
    Pinit [3] := chr (23);

    Pline [1] := chr (ESC);
    Pline [3] := chr (Bytes mod 256);
    Pline [4] := chr (Bytes div 256);

    Plfl [1] := chr (ESC);
    Plfl [3] := chr (1);

    Pexit [1] := chr (ESC);
  end; {with}

  Dump (DumpParams)
end {Print}.
```

{Listing 3: Program Print, Imagewriter-Version}

```
program Print; {Imagewriter-Version}
uses TurtleGraphics, SuperDump;

const ESC = 27;

var   CtrlBlock  : CtrlBlk;
      DumpParams : Dumptype;
      S          : string;
      Bytes      : integer;

begin {Print}
  Initturtle;
  moveto (Ø, Ø);
  pencolor (white);
  moveto (279, 191);
  moveto (Ø, 191);
  moveto (Ø, Ø);

  SetDefault (DumpParams);
  DumpParams.CtrlAdr  := Adr (CtrlBlock);
  DumpParams.Needle   := 1;
  DumpParams.Comptype := 1; {fuer Apple IIe mit SSC}
 {DumpParams.Comptype := 2; {fuer Apple IIc mit SSC} }
  GetGrafBytes (DumpParams, Bytes, S);

  with CtrlBlock do {Imagewriter}
  begin
    Pinit := '-T16->';
    Pline := concat ('-n-G', S);
    Plfl  := '';
    Pexit := '-<-A';

    Pinit [1] := chr (ESC);
    Pinit [5] := chr (ESC);

    Pline [1] := chr (ESC);
    Pline [3] := chr (ESC);

    Pexit [1] := chr (ESC);
    Pexit [3] := chr (ESC);
  end; {with}

  Dump (DumpParams)
end {Print}.
```

peeker Börse

Gelegenheitsanzeigen

Sie können unter dieser Rubrik zu einem besonders günstigen Preis

- Ihre Hardware und Software verkaufen
- Ihre Hard- und Software suchen
- Kontakte knüpfen und vieles mehr

Musteranzeige privat

1 Druckzeile à 32 Buchstaben nur DM 5,50 zuzügl. ges. MwSt. Mindestens 2 Druckzeilen

Beispiel:

Verkaufe neuwertigen Typenraddrucker mit Apple-Interface.
Preis auf Anfrage. Tel. 007

nur DM 18,81 inkl. MwSt.

Musteranzeige gewerblich

1 Druckzeile à 32 Buchstaben nur DM 11,– zuzügl. ges. MwSt. Mindestens 2 Druckzeilen

Beispiel:

Neu im Angebot: Professionelle, separate Tastatur für Apple II plus 16 Funktionstasten und separatem Ziffernblock.
Fa. Keyboard & Co.

nur DM 62,70 inkl. MwSt.

peeker

Peeker-Börse

AUFTRAG FÜR KLEINANZEIGEN

Bitte veröffentlichen Sie in der nächsterreichbaren Ausgabe nachstehenden Text unter folgender Rubrik:

☐ suche Hardware ☐ suche Software ☐ Verschiedenes
☐ biete Hardware ☐ biete Software ☐ Chiffre

Bitte jeweils 32 Buchstaben pro Zeile – einschließlich Satzzeichen und Wortzwischenräume. Bitte Absender nicht vergessen. Mindestens 2 Zeilen. Chiffregebühr DM 6,– zuzügl. MwSt.

peeker

Produkt-Karte

Zu der im **Peeker**, Heft _____, Seite _____ erschienenen

Anzeige über _____

_____

bitte ich um detaillierte Information.

Ich wünsche ☐ Prospekt, Datenblatt ☐ Preisliste ☐ schriftliches Angebot ☐ tel. Rückruf

Menge	Produkt und Bestellnummer	à DM	gesamt DM

gebe ich nebenstehende Bestellung unter Anerkennung Ihrer in der Anzeige genannten Liefer- und Zahlungsbedingungen auf.

_____
Unterschrift (für Jugendliche unter 18 Jahren der Erziehungsberechtigte)

peeker

Info-Karte

Schreiben Sie uns, wenn Sie technische Fragen zu Aufsätzen im **Peeker** haben. Beachten Sie jedoch, daß die Redaktion keine Auskünfte über Bezugsquellen erteilt:

_____

_____

_____

_____

_____

_____

Peeker-Börse

Vorname, Name

Firma

Straße

Wohnort

PLZ/Ort

Bitte veröffentlichen Sie den umstehenden Text von _____ Zeilen à _____ DM in der nächsterreichbaren Ausgabe vom **Peeker**

Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze

Datum Unterschrift

Produkt-Karte

Karte bitte vollständig ausfüllen

Vorname, Name

Firma

Straße

PLZ/Ort

Telefon mit Vorwahl

Anschrift der Firma angeben, bei der Sie bestellen bzw. von der Sie Informationen wünschen

Info-Karte

Karte bitte vollständig ausfüllen

Vorname, Name

Firma

Straße

PLZ/Ort

Telefon mit Vorwahl

POSTKARTE

Peeker-Börse
Anzeigen-Service

Dr. Alfred Hüthig Verlag

Postfach 10 28 69

6900 Heidelberg 1

POSTKARTE

Inserent

Straße

PLZ/Ort

POSTKARTE

Peeker
Redaktion

Dr. Alfred Hüthig Verlag

Postfach 10 28 69

6900 Heidelberg 1

Produkt-Karte

Wünschen Sie weitere Informationen zu einer der im Heft erschienenen Anzeigen?

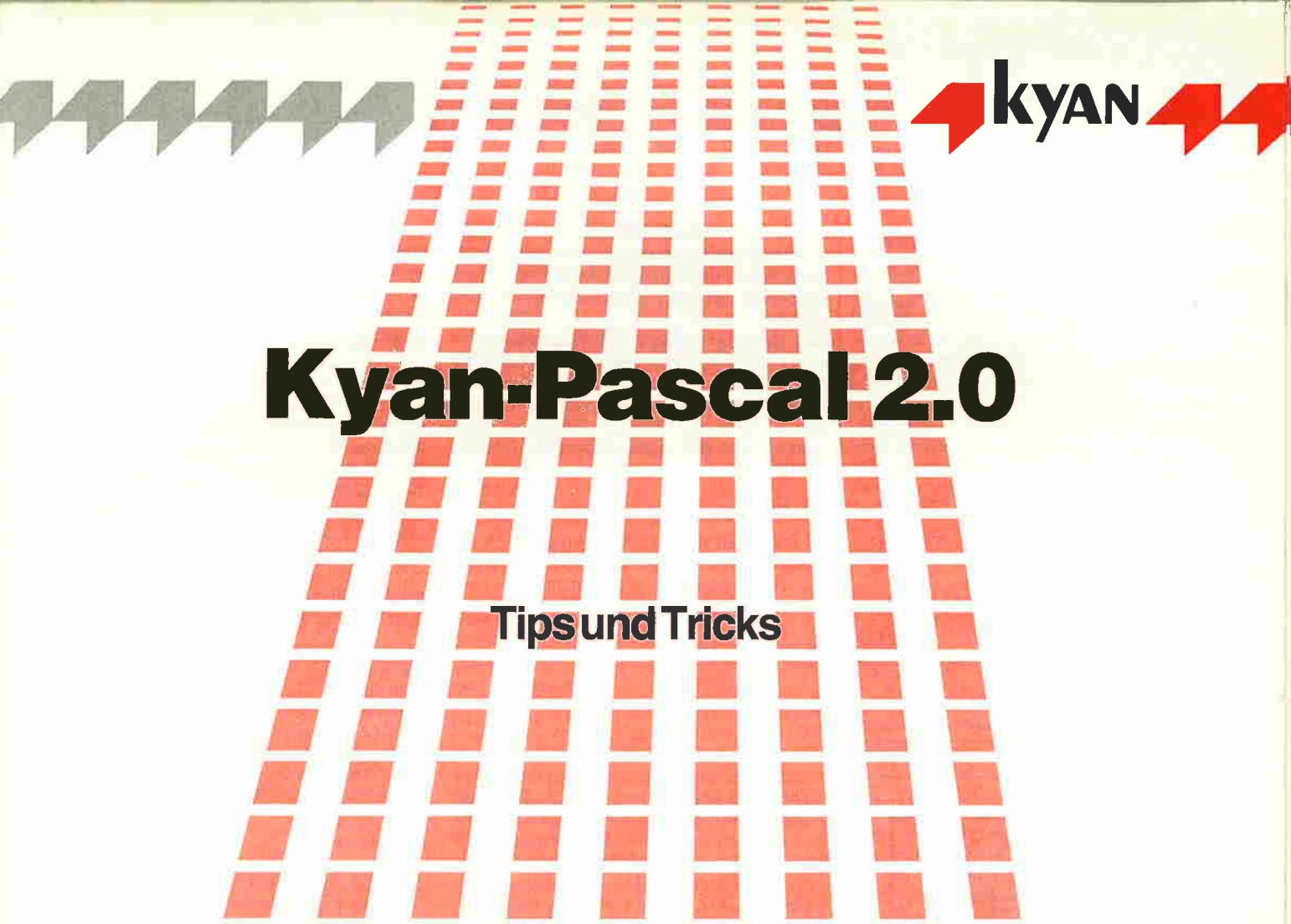
Nichts einfacher als das. Produkt-Karte ausfüllen, frankieren und an den Inserenten (nicht an die Peeker-Redaktion) senden.

Vorher aber nicht vergessen:
Kreuzen Sie an, welchen Informationswunsch Sie haben.

Damit erleichtern Sie dem Hersteller eine gezielte Beantwortung Ihrer Anfrage

Zum Schluß tragen Sie auf der Rückseite die genaue Anschrift des Inserenten und Ihren Absender ein.

PEEKER



Kyan-Pascal 2.0

Tips und Tricks

von Ulrich Stiehl

1. Kinderkrankheiten

Nachdem ich mich nun seit mehreren Wochen mit der Version 2.0 von Kyan-Pascal befaßt habe, kann ich nicht umhin, auf einige Unzulänglichkeiten hinzuweisen. Berücksichtigt man, daß man für ca. DM 170,- einen ca. 420 Seiten dicken Ordner sowie 2 beidseitig bespielte Disketten mit dem eigentlichen Kyan-Pascal, dem Kix-System und dem kompletten ProDOS (PRODOS 1.1.1 einschließlich BASIC.SYSTEM 1.1) erhält, welch letzteres allein ca. DM 170,- kostet, dann leuchtet ein, daß man viel Ware fürs Geld bekommt, insbesondere wenn man an Apple-UCSD-Pascal denkt, das nach wie vor ca. DM 900,- kostet. Turbo-Pascal 3.0 muß man gesondert sehen, denn es kostet zwar mit ca. DM 280,- nur ca. DM 100,- mehr als Kyan-Pascal 2.0, doch muß man die Z80-Karte mit CP/M für ca. DM 1000,- gesondert erwerben. Kyan 2.0 ist mithin das preiswerteste Pascal-System für den Apple II. Bei der nachfolgenden Kritik muß man also stets im Auge behalten, daß man von einem derart günstigen Pascal-System keine Wunder erwarten kann.

Der in vielerlei Hinsicht professionelle Eindruck, den Kyan-Pascal 2.0 hinterläßt, wird durch eine Reihe von elementaren Schlampereien getrübt. Die Kyan-Entwickler kommen mir manchmal vor wie hervorragende Dichter, die der Rechtschreibung entraten. Bei dem Vers

The seed you sow
another reaps,
the wealth you find
another keeps

sind Form und Inhalt in Harmonie. Überspitzt formuliert, hätten die Kyan-Leute hier gedichtet

The seed you
though another reaps,
the wells you find
another keeps

womit der Spruch völlig entstellt worden wäre.

1.1. Benutzerkomfort

Der Benutzerkomfort von Kyan-Pascal läßt manches zu wünschen übrig. Einige Beispiele:

– Der Editor ED nimmt korrekterweise nur Textfiles an, doch den Modulen PC und AS kann man auch Binärfiles anbieten, womit natürlich nur Schrott compiliert bzw. assembliert wird. Es hätte nur weniger Bytes bedurft, um diesen Fehler über die ohnehin benötigte Get-File-Info-Routine abzufangen.

– Der Assembler AS läßt eine Korrektur zu, wenn man sich vertippt hat, die Module ED und PC hingegen nicht. Gibt man z.B.

PC ---

ein, dann wird erst der immerhin 55 Blöcke umfassende Pascal-Compiler eingeladen und sodann prompt wieder verlassen. Das gleiche gilt für alle Kix-Module. Ich verkenne nicht, daß etwa CP/M dieselben Schwächen hat, z.B. bei der Eingabe von

PIP ---

Doch wenn man bedenkt, daß heute bereits Computer wie etwa der Macintosh gebaut werden, die selbst Lesebehinderte auf Bedienungsfehler aufmerksam machen können, dann scheint ein etwas größerer Komfort beim Error-Handling angebracht. Bei den nicht gerade kompakten Kyan-Modulen müßte hierfür noch Platz sein.

1.2. Apple-Versionen

Es steht außer Zweifel, daß Kyan-Pascal 2.0 vor dem Verkauf nicht auf den angegebenen Gerätekonfigurationen getestet worden ist. Einige Beispiele:

– Jedermann außer den Kyan-Entwicklern weiß, daß eine Videx-Karte keinen Home-Befehl (JSR $FC58) akzeptiert, sondern statt dessen Ctrl-L verlangt. Folglich sehen Apple-II-Besitzer mit einer Videx-Karte gelegentlich Schrott auf dem Monitor, weil der Bildschirm nicht gelöscht wird. Die Firma Apple selbst kümmert sich zwar auch nicht um Ctrl-L, wie jeder Videx-Besitzer nachprüfen kann, wenn er unter dem BASIC.SYSTEM den BYE-Befehl aufruft, doch dies steht auf einem anderen Blatt.

– An Apple-II-Besitzer ohne Kleinschreib-Umrüstsatz wurde nicht gedacht, denn alle Menü-Texte erscheinen in Kleinschreibung.

– Verschiedene Kix-Module, z.B. LPR, kann man, wenn eine Datei nicht existiert, nur verlas-

APPLE & CP/M-80 & MS-DOS SOFTWARE & HARDWARE

z.B. für APPLE II und Kompatible

Wir liefern die RAM-Karte (AE) für den Apple IIe mit max. 3 MB (Appleworks mit mehr als 2 MB)! 64-K-Ausf.	DM 650,–
Spee Demon 3,56 MHz Coproz. für II+/e (McT)	DM 700,–
Anpassung für APPLEWORKS 1,2 auf dem II+/e Original APPLE oder mit externer Tastatur in deutsch für SATURN 128K und IBS AP 33 1 MB!!	DM 170,–
UPC-Programmer-Card 2716-128 komfortabel	DM 580,–
72 I/O Port Card programmierbar DOS+CP/M	DM 350,–
AD 16 Ch. 12 Bit, schnell! (Applied Eng.)	DM 1150,–
PKASO/U-Printer-Karte (IS)	DM 550,–
CP/M-Plus-Card, 6 MHz, 64K, CP/M 3,0 (ALS)	DM 1150,–
Timemaster II H. O., die Uhrenkarte!(AE)	DM 540,–
ELF kompl. Statistik-Software (TWG)	DM 800,–
Prime-Plotter-Grafik-Software (Primesoft)	DM 900,–
Z-RAM 512 K für APPLE IIc (AE)	DM 1250,–

z.B. für IBM und Kompatible

APPLE Turnover (Vertex) Lesen/Schreiben von Apple Disks im IBM PC & Komp.	DM 1200,–
XENO-COPY plus (Vertex) Lesen/Schreiben div. CP/M & MS-DOS Formate im IBM	DM 600,–
ELF PC kompl. Statistik Software (TWG)	DM 800,–
PROM Blaster 28-Pin (Apparat Inc.)	DM 620,–

z.B. für alle Systeme

Printerchanger 3 parall. Drucker auf 1 Micro inkl. Kabel/Netzteil (Keyzone)	DM 570,–
Printersharer 3 Micros auf 1 parall. Drucker inkl. Kabel/Netzteil (Keyzone)	DM 460,–
Shufflebuffer 64K (IS)	DM 1250,–

Wir sind Import-Spezialisten und bieten Ihnen eine große Auswahl an Software und Hardware bedeutender Hersteller aus den USA und England Informationen gegen DM 3,– in Briefmarken.

WEISS COMPUTER Dipl.-Psych. Karl-Heinz Weiß
Am Wiesenhof 17, 2940 Wilhelmshaven, Tel. 04421/183179

Z80+ Card

Die leistungsfähige und vielseitige Z80 Karte für Ihren Apple II

Hardware:
+ ALS CP/M-Card- und (!) SoftCard-kompatibel.
+ 8MHz Taktfrequenz (ohne Waitzyklen) – dadurch werden Programme bis zu 4x schneller als bisher (2MHz Taktfrequenz im SoftCard-Modus).
+ Interrupts sind für die beiden Betriebsmodi der Z80+ Card getrennt zu- und abschaltbar.

Software:
+ CP/M 2.20/2.23 Pseudodisktreiber, der die eigenen 64K RAM der Z80+ Card im SoftCard-Modus nutzt.
+ CP/M-Patch generiert aus Ihrem SoftCard CP/M 2.20 ein CP/M 2.2 für den schnellen Taktmodus der Z80+ Card mit überragenden Eigenschaften: Bis zu 4x schnellerer Disk-Zugriff (Read), integrierte Pseudodisk mit 32/48/110K bei einem Apple mit 48/64/128K RAM, Unterstützung von Disk-Laufwerken bis 640K an den verschiedensten Controllern, Unterstützung der gängigsten RAM-Karten als Pseudodisk, schnelle Bildschirmausgabe, bedienerfreundlich durch neue und erweiterte Befehle in der Kommandoebene, 58K TPA.
+ Der Preis: 696,-- DM (= 610,53 DM ohne Mwst.)

Uwe Zimmermann – Microcomputerentwicklungen
Schwalbenstraße 30 – 6090 Rüsselsheim
Telefon: 06142 / 56 34 56

sen, wenn man die Esc-Taste drückt. Dies geht jedoch bekanntlich nicht beim IIe wegen eines Bugs in den ursprünglichen ROMs. Auch dies wurde von den Kyan-Entwicklern niemals ausprobiert.
Aus all diesen Gründen hätte man deshalb in der Kyan-Anleitung klarstellen müssen, daß einige Module nur auf dem Apple IIe mit den neuen ROMs funktionieren. (Neueste Information lt. Telex vom 15.5.86: Die Firma Videx kümmert sich jetzt selbst um die Videx-Kompatibilität.)

1.3. Debugging

In der Anleitung heißt es vollmundig „Kyan Pascal has been tested for conformance to the ISO Pascal standard". Dies stimmt nicht. Zwei Beispiele:

```
PROGRAM CONSTBUG;
CONST
C = 'A';
BEGIN
WRITELN (ORD(C))
END.
```

Selbst ein Primitivtest in der Art des Programms KYANKURS (Peeker, Heft 2/85, S. 61ff.), das jeden der Grundbefehle exemplifiziert, hätte sofort offenbart, daß der Compiler allen Char-Konstanten den Wert CHR(1) zuweist. Das Demo CONSTBUG ergibt immer 1, gleich welches Zeichen man C zuweist. Folglich funktioniert auch der CASE-Befehl mit Char-Konstanten nicht. WRITELN ('A') funktioniert natürlich. Würde man in P.OUT an den entsprechenden Stellen _lcw 1 durch _lcw + Ascii-Code der Char-Konstanten ersetzen, dann würde der CASE-Befehl später in A.OUT korrekt funktionieren. Der Bug liegt also am Compiler.

```
PROGRAM REFBUG;
VAR
I,J: FILE OF INTEGER;
BEGIN
{Mehrfaches Rewrite wäre okay}
REWRITE (I, '/RAM/I'); {Ref-No 1}
REWRITE (J, '/RAM/J'); {Ref-No 2}
{Mehrfaches Reset geht nicht}
RESET   (I, '/RAM/I'); {Ref-No 1}
RESET   (J, '/RAM/J'); {Ref-No 2}
{Hier Abbruch bei END-Routine}
END.
```

Auch diesen sog. File-Reference-Bug, der z.B. bei dem Programm FILEDEMOS (Peeker 3/86, S.44) oder dem obigen REFBUG zutage tritt, hätte man mit Primitivtests finden müssen. Ich vermute, daß der Bug in der Library LIB zu suchen ist.

```
_I EQU 22  ;_U EQU 22
```

In dem Include-File HIRES.I wurde statt _I fälschlicherweise _U getippt. Auch dieses Modul ist folglich vor dem Verkauf von Kyan 2.0 nicht mehr ausprobiert worden.

Nun muß man natürlich wissen, daß auch Turbo-Pascal 2.0 und Apple-Pascal 1.1 ursprünglich zahlreiche Bugs aufwiesen, die erst in späteren Versionen nach und nach beseitigt wurden. So enthält etwa das Addendum zu Apple-Pascal 1.2 auf den Seiten 77-81 eine lange Liste von Fehlern, die die Apple-Pascal-Entwickler selbst gefunden haben, von den nicht-erwähnten Unzulänglichkeiten ganz zu schweigen (z.B. Stack-Overflow-Bug u.a.). Erfreulicherweise arbeitet man bereits an der Kyan-Version 2.0.2, die u.a. die beschriebenen Fehler beheben soll.

2. Reset-Tips

Wenn man unter Kyan-Pascal Ctrl-Reset drückt, wird – wie auch in UCSD- und Turbo-Pascal – neu gebootet. Dies ist lästig, weil dadurch der Inhalt der 64K-RAM-Disk des Apple IIe/c von ProDOS frisch initialisiert wird. Es bieten sich zwei Lösungsmöglichkeiten an:
a) Man patcht das sog. RAM-Disk-Initflag (s. „A.Schäpers, ProDOS-Analyse", S. 297), wovon jedoch im allgemeinen abzuraten ist.
b) Man baut in sein Pascal-Programm die folgende Prozedur ein, die man im laufenden Programm *einmal* aufruft.

```
PROGRAM RESETDEMO;
VAR C:CHAR;
{-----------------------------}
PROCEDURE RESETVEKTOR;
BEGIN
#A
      JMP BOOT3
BOOT1 CLD
      JSR $BF00   ;Reboot-Befehl
      DB  $65
      DW  BOOT2
BOOT2 DB  $04
;
BOOT3 LDA #>BOOT1 ;Reset-Vektor
      STA $03F2
      LDA #<BOOT1
      STA $03F3
      EOR #$A5
      STA $03F4
#
END;
{-----------------------------}
BEGIN
WRITELN ('Ctrl-Reset druecken!');
RESETVEKTOR; READLN (C)
END.
```

Um diese Prozedur zu verstehen, muß man etwas über die **Restart-Routine** wissen, die als Bestandteil des KIX.SYSTEMs in die Language-Card ab $D100ff. geschoben wird, wo sich zuvor die **Reboot-Routine** von PRODOS befand. Wenn man über das ProDOS-MLI (siehe obige Reset-Prozedur) oder unter dem BASIC.SYSTEM über den BYE-Befehl die Reboot-Routine aufruft, so wird der Reset-Vektor auf $1000 gelegt, die Reboot-Routine selbst nach $1000 geschoben und die Eingabe eines SYS-Files erwartet, der dann geladen wird, z.B. der FILER oder das BASIC.SYSTEM. Bitte probieren Sie zunächst den BYE-Befehl aus, falls Sie ihn noch nicht kennen sollten.
Verläßt man nun ein Kyan-Programm über die eigentlich erwartete Reboot-Routine, so wird in Wirklichkeit die Restart-Routine von $D100ff. nach $1000ff. geschoben. Diese Restart-Routine enthält selbst ein 512 Bytes umfassendes Menü-Programm, das nach $BC00-$BDFF geschoben und über JMP $BC00 gestartet wird. Es erscheint dann der übliche %-Prompt, und man kann nun ED, PC, FILER, BASIC.SYSTEM usw. laden.

Setzt man nun in einem eigenen Kyan-Programm den Reset-Vektor dergestalt, daß nach Ctrl-Reset scheinbar Reboot ausgeführt wird, so gelangt man in der geschilderten Weise wieder zum %-Prompt, wie dies auch automatisch der Fall ist, wenn ein Kyan-Programm regulär über „END." verlassen wird.

Zusammenfassung
(1) Das KIX.SYSTEM verschiebt Restart nach $D100ff. und entfernt damit das ursprüngliche Reboot.
(2) Der MLI-Reboot-Befehl setzt den Reset-Vektor auf $1000 und verschiebt Restart nach $1000.
(3) Restart enthält das Menü-Programm und verschiebt es nach $BC00-$BDFF und springt dann selbst nach $BC00, womit der %-Prompt erscheint.

(a) Ein auf den MLI-Reboot-Befehl gerichteter Reset-Vektor im eigenen Pascal-Programm initiiert nach Ctrl-Reset die Routinen (2) und (3).
(b) Drückt man jedoch danach erneut Ctrl-Reset, so wird gebootet, weil die Restart-Routine den Reset-Vektor zerstört. Man könnte dies patchen, wie im Kyan-Rundschreiben #2 von einem Club-Mitglied angegeben wurde, doch funktioniert dies nur dann, wenn gerade ein eigenes Pascal-Programm über „END." verlassen worden ist. Nach PC und AS geht dies nicht, weil diese Programme aus unerfindlichen Gründen je eine Kopie vom späteren $BC00-$BDFF-Inhalt enthalten, der *ohne* die Restart-Routine direkt dorthin verschoben wird. PC und AS werden also mit JMP $BC00 verlassen.

3. Bit-Map-Tips

Die System-Bit-Map markiert diejenigen Pages (Speicherseiten), die für ProDOS tabu sind, d.h. in die keine Daten von Diskette geladen werden dürfen. Entgegen meiner ursprünglichen Vermutung gehört es zu den Schlampereien des KIX.SYSTEMs, daß es die Bit-Map nicht ändert oder wenigstens normalisiert, was sonst bei jedem SYS-Programm der Fall ist.
Die ProDOS-Reboot-Routine würde den Speicher $0800-$BEFF (sowie darüber hinaus $0200-$03FF) automatisch freimachen, doch sieht die Kyan-Restart-Routine hierfür nichts vor. Booten Sie eine ProDOS-Systemdiskette mit dem BASIC.SYSTEM, legen Sie die Kix-Diskette ein und machen Sie dann folgendes Experiment:
Ü BLOADKIX.SYSTEM,A$2000,TSYS
Ü CALL-151
* 1000:0
* 1001<1000.10FFM
* 0200<1000.10FFM
* 2000G
% PROGRAM
„PROGRAM" sei ein beliebiges Programm. Sie erhalten nun mit Gewißheit die MLI-Fehlermeldung $56 (= System-Bit-Map-Konflikt), weil PROGRAM versucht, LIB in den Bereich $9000-$BEFF einzuladen, was jedoch das MLI nicht zuläßt, weil die Bit-Map noch die Belegung des BASIC.SYSTEMs aufweist. Sie werden sich nun jedoch fragen, weshalb man trotzdem das KIX.SYSTEM vom BASIC.SYSTEM aus mit
Ü -KIX.SYSTEM
starten kann. Des Rätsels Lösung: Der Strich-Befehl normalisiert die Bit-Map, wenn er einen SYS-File vorfindet (gilt nicht für BIN-Files usw.)

Um nun diesen Bit-Map-Fehler zu beheben, verankern wir in der Kyan-Library namens STDLIB.S folgende Routine, wobei wir bei Bedarf gleichzeitig noch den List-off/on- und evtl. den Macros-off/on-Befehl einfügen, damit der Assembler AS im Falle der List-Option (-L) nicht immer die stets gleichbleibende Library listet. Danach können wir jedes compilierte Kyan-Programm, gleichviel ob BIN- oder SYS-File, vom BASIC.SYSTEM aus mit
-PROGRAM
starten, denn die Bit-Map-Normalisierung wird direkt vor dem Laden der Kyan-Library (LIB) ausgeführt.

```
;-----------------------------
 LST OFF ;STDLIB.S nicht listen
_lib equ $9000    ;folgt LST OFF
;-----------------------------
; usw. bis.....
;-----------------------------
 LST ON     ;jetzt wieder listen
 MEX OFF    ;ohne Makros
_start ldx #$ff ;Programmanfang
 txs
;-----------------------------
;Bit-Map tabu: $0000-$01FF +
;$0400-$07FF + $BF00-$BFFF
 CLD
 LDA #0
 LDX #$59
BM00 STA _MLI,X
 INX
 CPX #$6F
 BNE BM00
 LDA #$CF    ;%11001111
 STA $BF58
 LDA #$01    ;%00000001
 STA $BF6F
;-----------------------------
;Ab hier normal weiter:
 jsr _LoadPascalLibrary
```

4. Präfix-Tips

Die Präfix-Konventionen von Kyan-Pascal dürften Anfänger und Fortgeschrittene gleichermaßen verwirren, denn es fehlt ihnen eine konsequente Linie. Leider schweigt sich das Manual über die Präfix-Regeln aus, so daß wir etwas weiter ausholen müssen. Wir gehen dabei für unsere Erörterungen von den Prämissen aus, daß
1. ein Subdirectory namens /RAM/BIN und
2. ein momentanes Präfix (= Ersatz-Präfix = Default-Präfix) namens /K existieren.
Man beachte dabei, daß /RAM/BIN keine RAM-Disk sein muß, denn Analoges gilt auch für Festplatten und Disketten mit dem Subdirectory /RAM/BIN, wobei jedoch /RAM ein Volume-Directory sein muß. Die Regeln gelten also nicht für /K/RAM/BIN usw. Ferner beachte man, daß das willkürlich gewählte Ersatz-Präfix /K auch z.B. /K/SUBDIR o.ä. heißen könnte, d.h. es muß kein Volume-Directory sein. Doch vorab die Präfix-Konventionen des BASIC.SYSTEMs unter den obigen Prämissen:
(a) Wenn man „-STARTUP" eingibt, so entspricht dies dem Befehl „-/K/STARTUP", d.h. das Ersatz-Präfix wird automatisch hinzugefügt. Wird nun STARTUP nicht in /K gefunden, so erfolgt direkt die Fehlermeldung „File not found"; eine weitergehende Suche in anderen Directories findet nicht statt.
(b) Wenn man „-/K/STARTUP" eingibt, so ist das Ersatz-Präfix mit dem vorgegebenen Präfix identisch, so daß die in (a) beschriebene Suchroutine initiiert wird.
(c) Wenn man „-/RAM/BIN/STARTUP" eingibt, so werden *alle* angeschlossenen Volumes nach dem vorgegebenen Präfix /RAM/BIN abgesucht, und eine Fehlermeldung erfolgt erst dann, wenn /RAM/BIN in keinem Volume-Directory enthalten ist.
(d) Wenn man „-BIN/STARTUP" eingibt, so wird das Subdirectory /K/BIN nur in /K gesucht, also nicht etwa zusätzlich in /RAM.
Fazit: Ein vollständiger Pfadname, der mit „/" beginnt, wird in allen Volumes gesucht, ein unvollständiger Pfadname, der nicht mit „/" beginnt, dagegen nur im Ersatz-Präfix. Dabei bezeichnen wir als *Präfix-Directory* dasjenige Directory, das durch das vorgegebene Präfix (bei -/RAM/BIN/STARTUP → /RAM/BIN) oder das Ersatz-Präfix (bei -STARTUP → /K/) bestimmt ist. Den Sonderfall eines Null-Präfixes klammern wir hier aus, da in diesem Fall unter dem BASIC.SYSTEM die Ersatz-Slot/Drive-Parameter herangezogen werden.

Nun zu den Präfix-Konventionen unter Kyan-Pascal:
● Wenn man im KIX.SYSTEM-Menü *ohne* vorgegebenes Präfix die Programmnamen
% ED (steht auch für PC, AS, BASIC.SYSTEM usw.)
% PROGRAM (steht für alle compilierten Kyan-Programme, die SYS- oder BIN-Files sein können)
% KIX (steht für alle Kix-Utilities, z.B. LS usw.)
eingibt, so werden diese Programme *zunächst* in /RAM/BIN gesucht, gleichviel ob das Ersatz-Präfix /RAM/BIN oder /K lautet. Werden die Programme im Subdirectory /RAM/BIN gefunden, so werden sie direkt von dort gestartet. Werden sie jedoch dort nicht gefunden, so werden sie im Ersatz-Präfix-Directory /K gesucht. Werden sie auch dort nicht gefunden, so erfolgt die „File not found"-Fehlermeldung.
● Wenn man *mit* vorgegebenem Präfix die Programmnamen
% /K/ED
% /K/PROGRAM
% /K/KIX
eingibt, so werden die Programme direkt vom Directory /K gestartet, falls sie dort enthalten sind. Andernfalls erfolgt die „File not found"-Fehlermeldung, d.h. im Subdirectory /RAM/BIN wird in diesem Fall nicht mehr gesucht.

4.1. ED und PROGRAM.P

Die Programme ED usw. laden meist weitere Dateien nach, so daß auch für diese Dateien die Präfix-Konventionen zu erlernen sind.
ED (= Kyan-Editor) lädt den spezifizierten Textfile (mit Suffix .P), also PROGRAM.P, aus dem Präfix-Directory nach und speichert ihn nach Verlassen des Editors wieder dort zurück.
% ED PROGRAM.P
sucht zunächst ED im Subdirectory /RAM/BIN, falls dort nicht gefunden, in /K; falls dort auch nicht gefunden → Fehlermeldung. Dann wird PROGRAM.P von /K (und *nicht* zuerst von /RAM/BIN!) zu laden versucht; falls dort nicht gefunden → Fehlermeldung.
% ED /K/PROGRAM.P
sucht zunächst ED wie oben. Dann wird PROGRAM.P direkt von /K zu laden versucht; falls dort nicht gefunden → Fehlermeldung.
% /K/ED /K/PROGRAM.P
sucht ED direkt in /K und danach PROGRAM.P direkt in /K; falls nicht gefunden → Fehlermeldung.
Beim Verlassen des Editors über X (= Exit) wird PROGRAM.P in das Präfix-Directory zurückgeschrieben, falls man nicht über P (= Präfix)

einen anderen Pfadnamen (= Präfix-Directory + Dateinamen) spezifiziert.
Halten wir also fest: Der *erste* Programmname nach dem „%" wird immer zuerst in /RAM/BIN gesucht, falls kein Präfix *mit* Schrägstrich, also kein vollständiger Pfadname, angegeben wurde. Dagegen wird der *zweite* Programmname nach „%" niemals in /RAM/BIN gesucht, es sei denn, man würde z.B. ED /RAM/BIN/PROGRAM.P spezifizieren *oder* das Ersatz-Präfix wäre nicht auf /K, sondern auf /RAM/BIN eingestellt.

4.2. PC, P.OUT, AS, A.OUT

PC (= Pascal-Compiler) lädt PROGRAM.P nach, speichert dann P.OUT im Ersatz-Präfix-Directory (immer dort!), lädt bzw. „chained" dann den Assembler AS, der seinerseits die Standard-Library STDLIB.S lädt und am Schluß PROGRAM im Präfix-Directory oder A.OUT im Ersatz-Präfix-Directory (immer dort!) abspeichert.
% PC PROGRAM.P lädt zunächst PC von /RAM/BIN oder ersatzweise, d.h. falls dort nicht gefunden, von /K. Dann wird PROGRAM.P direkt von /K geladen. PC erzeugt und speichert jetzt P.OUT (= Assembler-Quelltext *ohne* Pascal-Quelltext-Zeilen) in /K. Dann wird AS von /RAM/BIN oder ersatzweise von /K geladen. AS lädt dann STDLIB.S von /RAM/BIN oder ersatzweise von /K. AS erzeugt und speichert jetzt PROGRAM in /K.
% /K/PC /K/PROGRAM.P lädt PC direct von /K, doch werden AS und STDLIB.S trotzdem zunächst von /RAM/BIN zu laden versucht. P.OUT wird in /K gespeichert.
% /K/PC /RAM/BIN/PROGRAM.P lädt PROGRAM.P direkt von /RAM/BIN und speichert PROGRAM direkt in /RAM/BIN, jedoch P.OUT in /K.

Andere Varianten
% PC PROGRAM.P > ERRORS würde lt. Manual zusätzlich einen Textfile mit Compiler-Fehlermeldungen in /K erzeugen. Dies ist jedoch nicht aus dem PC-Menü ersichtlich und funktioniert deshalb bei Kyan 2.0 auch nicht. Die Slot-Option
% PC PROGRAM.P >1
ist allerdings möglich (1 = Slot 1).
Achtung: Wenn der Pascal-Quelltext nicht das Suffix „.P" hat, dann wird ersatzweise A.OUT erzeugt, und zwar immer im Ersatz-Präfix.
% PC /RAM/BIN/QUELL erzeugt aus dem Textfile QUELL nicht den Binärfile QUELL, sondern A.OUT, und zwar nicht in /RAM/BIN, sondern in /K, wo sich später auch P.OUT befindet.
% PC QUELL -O QUELLOBJ erzeugt aus dem Quelltext QUELL den Objektcode QUELLOBJ in /K.

% PC PROGRAM.P -S erzeugt P.OUT (= Assembler-Quelltext *mit* Pascal-Quelltext-Zeilen) in /K. PC „chained" nicht AS; folglich wird PROGRAM nicht erzeugt. Man beachte, daß unter Kyan-Pascal 2.0 die S-Option aus PROGRAM.P nicht PROGRAM.S erzeugt, wie fälschlicherweise im Handbuch, III/5 erwähnt wird.
% AS P.OUT lädt AS von /RAM/BIN oder ersatzweise von /K, dann STDLIB.S von /RAM/BIN oder ersatzweise von /K und erzeugt schließlich den Objektcode A.OUT in /K. (A.OUT ist inhaltlich mit PROGRAM identisch.)
% AS P.OUT -O PROGRAM erzeugt anstelle von A.OUT in /K den Objektcode PROGRAM.
% AS P.OUT -L >LISTING erzeugt neben A.OUT in /K den Textfile LISTING (Assembler-Quelltext + Hexdump des Maschinenprogramms). Für die Bildschirmausgabe genügt
% AS P.OUT -L
Für die Druckerausgabe in Slot 1 lautet der Befehl
% AS P.OUT -L >1

Man beachte, daß P.OUT immer assembliert werden kann, gleichviel ob P.OUT über „PC PROGRAM.P" oder über „PC PROGRAM.P -S" erzeugt wurde. Ferner merken wir uns, daß P.OUT und A.OUT stets im Ersatz-Präfix gespeichert werden. Wenn man wünscht, daß P.OUT in einem vorgegebenen Subdirectory gespeichert wird, so muß dieses Subdirectory zugleich Ersatz-Präfix sein. Nehmen wir an, PC sei in /A, PROGRAM.P sei in /B und Ersatz-Präfix sei /C, das auch AS und STDLIB.S enthalte. Dann erzeugt der Befehl
% /A/PC /B/PROGRAM.P
P.OUT in /C und PROGRAM in /B.

4.3. PROGRAM und LIB

Der Quelltext STDLIB.S enthält die Makrodefinitionen für den Assembler sowie die Laderoutine für den Objektcode LIB, der in den Speicher ab $9000 geladen wird und die Runtime-Bibliothek darstellt. LIB ist also nicht etwa der Objektcode von STDLIB.S, sondern enthält die von allen compilierten Pascal-Programmen gemeinsam benötigten Routinen (z.B. Fließkomma-Mathematik usw.).

Speicherorganisation

An dieser Stelle ist ein kurzer Überblick über die Speicherorganisation unter Kyan 2.0 erforderlich, da die Abbildung im Peeker, Heft 4/86, S. 49 nur für Kyan 1.2 galt.
● Bevor das compilierte Pascal-Programm (PROGRAM) geladen wird, befindet sich das Menü-Programm (mit %-Prompt) im Bereich $BC00-$BDFF, gefolgt von der ProDOS-Global-Page ($BF00-$BFFF).
● Ein normales Kyan-Programm (PROGRAM) wird ab $0800 in den Speicher geladen und lädt dann seinerseits die Bibliothek LIB in den Speicher $9000-$BED3, also praktisch bis $BEFF, womit das Menü-Programm überschrieben wird, während die Global-Page natürlich erhalten bleibt. Für die Programmvariablen steht dann der Datenpool (= Heap) ab $8FFF abwärts bis zum Programmende (= LOMEM = Lowest Memory) zur Verfügung. Also:

```
$0800-LOMEM: PROGRAM
LOMEM-$8FFF: Pool (ab $8FFF rückwärts)
$9000-$BEFF: LIB
$BF00-$BFFF: Global-Page
```

● Wenn zu Beginn von PROGRAM.P mit
#A
_USESHIRES
#
der Speicher der HGR-Seite 1 ($2000-$3FFF) für Grafik reserviert wird, dann ergibt sich folgende Speicherverteilung:

```
$0800-$1FFF: Pool (ab $1FFF rückwärts)
$2000-$3FFF: HGR Seite 1
$4000-LOMEM: PROGRAM
$9000-$BEFF: LIB
$BF00-$BFFF: Global-Page
```

● Wenn zu Beginn von PROGRAM.P mit
#A
_SYSTEMFILE
#
ein SYS-File anstelle eines BIN-Files deklariert wird, so berührt dies die *spätere* Speicherverteilung nicht, d.h. SYS-Files sind sowohl bei Nicht-Grafik- als auch bei Grafik-Programmen möglich, wobei im letzteren Fall die Doppeldeklarierung
#A
_SYSTEMFILE
_USESHIRES
#
erforderlich ist. SYS-Files werden jedoch zunächst ab $2000 in den Speicher geladen und dann entweder nach $0800 (= Nicht-Grafik-Programm) oder nach $4000 (= Grafik-Programm) verschoben. Der Vorteil eines SYS-Files besteht darin, daß der erste Directory-SYS-File (mit Suffix „.SYSTEM") von einer ProDOS-Diskette direkt nach dem Booten gestartet wird.
● Wenn man ein Pascal-Programm über „END." verläßt, so wird wieder das Menü-Programm von $D100ff. nach $BC00ff. kopiert, und der untere Speicher ab $0800ff. ist dann vorübergehend frei bis zum nächsten Programmstart.
(Die ProDOS-I/O-Puffer – für jede geöffnete Datei $0400 Bytes – liegen übrigens im Datenpool, und zwar offenbar stets ab LOMEM modulo 256 aufwärts.)

% PROGRAM lädt PROGRAM von /RAM/BIN oder ersatzweise von /K und danach LIB von /RAM/BIN oder ersatzweise von /K.
% /K/PROGRAM lädt PROGRAM direkt von /K und danach LIB von /RAM/BIN oder ersatzweise von /K.

4.4. Weitere Feinheiten

Damit haben wir die wichtigsten Präfix-Konventionen von Kyan-Pascal kennengelernt. In Wirklichkeit sind die Zusammenhänge noch viel komplizierter. Nehmen wir an, es existieren /K, /K/BIN und /RAM/BIN:
(a) Wenn nun /K/BIN den Compiler PC enthält und /K den Quellcode PROGRAM.P und /K Ersatz-Präfix ist, dann findet der Befehl „% PC PROGRAM.P" trotzdem PC in /K/BIN, nachdem in /RAM/BIN vergeblich gesucht worden ist, und anschließend wird PROGRAM.P in /K gefunden und P.OUT sowie PROGRAM werden in /K gespeichert.
(b) Wenn nun /K/BIN den Objektcode PROGRAM enthält und /K die Library LIB und /K Ersatz-Präfix ist, dann findet der Befehl „% PROGRAM" zwar PROGRAM in /K/BIN, aber danach nicht mehr LIB in /K, obwohl das Ersatz-Präfix /K lautet. Erklärungsversuch: Zunächst wird /RAM/BIN/PROGRAM gesucht; Suche erfolglos. Dann wird /RAM abgehackt und durch /K ersetzt, womit /K/BIN/PROGRAM entsteht. Nun wird /K/BIN/PROGRAM mit Erfolg gesucht, aber das Ersatz-Präfix heißt jetzt für Kyan-Pascal /K/BIN, und in /K/BIN ist LIB nicht enthalten.
All dies ist für den Anfänger natürlich völlig undurchsichtig. Aber es geht noch weiter. Wir haben bislang unterstellt, daß ED, PC, AS, PROGRAM usw. automatisch zunächst in

/RAM/BIN gesucht werden. Dies trifft jedoch nur dann zu, wenn die Speicherstelle $BF98 in der Global-Page (sog. Machine-Identification-Code) einen 128K-Apple ausweist (= Apple IIe mit 64K-Karte oder IIc). Bei einem Apple IIe ohne 64K-Karte oder einem Apple II+ wird /RAM/BIN nicht berücksichtigt, selbst wenn eine große RAM-Karte oder gar eine Festplatte ein Subdirectory /RAM/BIN aufweisen. Daraus ergeben sich zwei Manipulationsmöglichkeiten:

(a) Wenn man als Besitzer eines Apple II+ mit RAM-Karte oder Festplatte darauf Wert legt, daß ein Subdirectory namens /RAM/BIN direkt angesprochen wird, so setze man Bit 4 von $BF98. Bei einem II+ mit 80-Zeichenkarte und ohne Hardware-Uhr lautet das Byte dann $B2.
(b) Wenn man als Besitzer eines Apple IIe mit 64K-Karte wegen einer Festplatte o.ä. nicht will, daß das Subdirectory /RAM/BIN der 64K-RAM-Karte angesprochen wird, so lösche man Bit 4 von $BF98. Bei einem IIe mit 80-Zeichenkarte und ohne Hardware-Uhr lautet das Byte dann $A2. Vorteil: Durch die Suche nach /RAM/BIN geht unnötig viel Zeit verloren. Wenn man /RAM in /REM umbenennt, merkt man dies sehr deutlich, denn dann werden alle Volumes vergeblich nach /RAM abgesucht.

5. Compiler-Tips

Viele besitzen weder eine große RAM-Karte noch eine Festplatte, sondern müssen mit einem einzigen Laufwerk und evtl. einer 64K-RAM-Karte vorlieb nehmen. Wie man unter diesen Umständen trotzdem vernünftig arbeiten und sogar die Compilierungs- und Assemblierungszeiten unter Kyan-Pascal um über 50% reduzieren kann, werde ich Ihnen jetzt zeigen.

Variante 1

Besitzer eines Apple II+ oder eines IIe ohne 64K-Karte stellen sich folgende 140K-Arbeitsdiskette zusammen:

```
/RAM
DIR BIN
/RAM/BIN
     SYS KIX.SYSTEM
     BIN PC
     BIN AS
     TXT STDLIB.S
     BIN ED
     BIN LIB
     BIN FID
     TXT PROGRAM.P
     BIN PROGRAM
     TXT P.OUT
```

(a) Booten Sie von einer ProDOS-Systemdiskette, z.B. von Kyan-Disk 2, Seite 2.
(b) Legen Sie die Arbeitsdiskette ein, setzen Sie das Präfix mit
PREFIX/RAM/BIN
auf das Subdirectory und starten Sie dann das KIX.SYSTEM mit
-KIX.SYSTEM
Für die Arbeitsdateien PROGRAM.P usw. bleiben dann noch über 100 Blöcke frei, und beim Compilieren und Assemblieren wird direkt auf das Subdirectory /RAM/BIN zugegriffen.
Zu Testzwecken wurde mit dieser Konfiguration ein bestimmtes Programm mit
% PC PROGRAM.P
compiliert, wobei sich die Compilierungs- und Assemblierungszeit auf fast genau 100s (= 100%) belief.

Variante 2

Zur Abschreckung wird die folgende „Lösung" vorgeführt, die sich automatisch dann ergibt, wenn man einen IIc oder IIe mit 64K-Karte besitzt und so vorgeht, wie es das Kyan-Manual beschreibt. Danach bootet man die Kyan-Disk 1, Seite 1 (mit Präfix /KYAN.PASCAL), womit das KIX.SYSTEM gestartet wird, das automatisch /RAM/BIN erzeugt und die Dateien STDLIB.S, AS, ED, MENU, INTRO, QUIT, KIX und CD in /RAM/BIN kopiert. Nehmen wir an, man würde jetzt die Arbeitsdiskette /K einlegen, die neben PC auch FID und PROGRAM.P usw. enthielte. Dann würde das Testprogramm mit dem Befehl
% /K/PC /K/PROGRAM.P
in ca. 85s (= 85%) compiliert, wenn man /K mit angibt oder das Ersatz-Präfix zuvor auf /K eingestellt hat und dann
% PC PROGRAM.P
eingibt. Das Präfix kann nicht auf /RAM/BIN eingestellt werden, weil dort nicht genügend Platz für P.OUT wäre. Entfernt man auf /RAM/BIN die Dateien ED, MENU usw. und kopiert statt dessen PC auf /RAM/BIN, womit sich also PC, AS und STDLIB.S, die bei der Compilierung benötigt werden, in /RAM/BIN befinden, so reduziert sich die Compilierungszeit auf ca. 82s (= 82%). Es nützt also nicht viel, wenn PC, AS und STDLIB.S in einer RAM-Disk liegen, weil das Einlesen dieser drei Dateien zusammen nur ca. 8-10s benötigt, wie man vom BASIC.SYSTEM aus über den BLOAD-Befehl leicht feststellen kann.

Variante 3

Nun zeigen wir eine erheblich bessere Lösung für Besitzer einer 64K-Karte, die die Compilierungszeit gegenüber Variante 2 um über 50% reduziert. Zu diesem Zweck legen wir folgende Arbeitsdiskette an:

```
/K
SYS KIX.SYSTEM
BIN ED
BIN LIB
BIN PC
BIN FID
TXT PROGRAM.P
BIN PROGRAM
/K/BIN
   BIN AS
   TXT STDLIB.S
   TXT P.OUT
```

(a) Booten Sie zunächst eine ProDOS-Systemdiskette, legen Sie /K ein und setzen Sie das Präfix mit PREFIX/K auf /K.
(b) Starten Sie dann das KIX.SYSTEM, das in /K/BIN nur die Dateien AS und STDLIB.S vorfindet und diese in /RAM/BIN kopiert.
(c) Danach setzen Sie das Präfix auf /RAM/BIN. *Dies ist der Trick!* Wenn Sie nämlich jetzt mit
%/K/PC /K/PROGRAM.P
compilieren, so wird P.OUT in /RAM/BIN abgelegt, jedoch PROGRAM in /K. Die meiste Zeit verbraucht nämlich der Compiler für das Speichern von P.OUT sowie der Assembler für das Einlesen von P.OUT, während das Laden von PROGRAM.P und das Speichern von PROGRAM nur unwesentlich verbessert werden können. Die Compilierungszeit reduziert sich dann auf sage und schreibe 41s (= 41%). Dies entspricht einer 60%igen Verbesserung gegenüber Variante 1 und einer 50%igen Verbesserung gegenüber Variante 2. Hinzu kommt, daß Variante 3 der Datensicherheit dient, denn P.OUT ist eine „Wegwerfdatei", während PROGRAM.P und PROGRAM später noch benötigt werden. Da die RAM-Disk zudem nur AS und STDLIB.S enthält, kann P.OUT dort über 70 Blöcke groß werden.
Die Variante 3 ist derart effizient, daß man beispielsweise mit der Megaboard-Festplatte nur noch eine Reduzierung auf 29s (= 29%) erreicht.

6. FID statt Kix-Utilities

Die Kix-Utilities LS, CMP, LPR usw. sind teilweise sehr leistungsfähig, doch nehmen sie fast eine ganze 140K-Diskette in Beschlag. Deshalb habe ich für die Kyan-Club-Mitglieder ein sehr kompaktes Kyan-FID (analog zum DOS-3.3-FID) in 3 Versionen mit insgesamt folgenden Befehlen entwickelt:
(a) Quit, Online, Prefix, Catalog, Create, Delete, Lock, Unlock, Rename
(b) Bsave, Bload
(c) Copy
FID (= Komplettversion) enthält alle Befehle.
FID1 enthält die Befehle (a) und (c).
FID2 enthält die Befehle (a) und (b).
Andere Befehlskombinationen kann man sich anhand der Quelltexte FID.P, FID1.P und FID2.P selbst zusammenstellen. Alle Versionen enthalten neben dem Assemblerteil ein kleines Menü-Programm in Pascal. Im Gegensatz zu den Kix-Utilities können damit die FID-Routinen in eigene Pascal-Programme eingebaut werden. 1-Drive-Besitzer können FID1 benutzen, das als Objectcode nur 14 Blöcke einnimmt und damit auf der Arbeitsdiskette immer Platz findet. Nähere Informationen über FID, das nur an Club-Mitglieder abgegeben wird (Preis DM 20,-), können Sie dem Club-Rundschreiben #2 entnehmen.

7. 80-Zeichenkarte von Ueding

Die Firma Ueding in Menden hat uns freundlicherweise kurzfristig eine 80-Z/Z-Karte zur Überprüfung der Kyan-Probleme beim II+ ausgeliehen. Diese Karte, die über 2 Zeichensätze mit 9×10-Matrix (mit deutlich besserem Schriftbild als beim IIe!) verfügt, setzt einen Monitor voraus, bei dem man die Bildbreite (von außen) verstellen kann, ist nur für Slot 3 gedacht und scrollt recht schnell. ProDOS erkennt die Karte nicht, was jedoch für den Kyan-ED2 unerheblich ist. Verwendet man sie jedoch auf dem IIe statt auf dem II+, so müssen vorher die Softswitches normalisiert werden. Es genügt der folgende Patch:

```
#A
 STX  _T
;LDA  $BF98  ;Machine-ID
;ORA  #$02   ;80-Z/Z-Bit
;STA  $BF98  ;Machine-ID
 JSR  $FE89  ;IN#0
 JSR  $FE93  ;PR#0
 STA  $C006  ;Slot $C100ff.
 STA  $C00B  ;Slot $C300ff.
 JSR  $C300  ;Ueding-Karte an
 LDX  _T
#
```

Die Karte funktioniert dann mit dem Kyan-ED2, doch ist bei Return der Cursor nicht sichtbar. Die liegt jedoch an Kyan und nicht an der Ueding-Karte.

Adreßverwaltungsprogramm mit dBase II

Listing des überarbeiteten Quelltextes

von Ing. (grad.) Ernst Fischer

Im Peeker, Heft 8/1985, S. 40-44, ist ein Adreßverwaltungsprogramm beschrieben worden, das auf die Peeker-Sammeldiskette #8 aufgenommen, aber nicht als Quelltext abgedruckt worden ist. Da viele Leser das dBase-Programm ändern oder auf andere Rechner anpassen wollten, bringen wir nachfolgend den kompletten Quelltext einschließlich einiger *kursiver* Ergänzungen, die Sie bei Bedarf in die Disk #8 einarbeiten können. Beachten Sie, daß das Listing aus drucktechnischen Gründen insbesondere bei den SAY-Zeilen auf 65 Zeichen/Zeile umbrochen wurde; die Funktionsfähigkeit des Programms wird dadurch jedoch nicht berührt.

1. AD-START.CMD

```
* DO START.CMD (Startet das Adv)
ERASE
SET TALK OFF
RESTORE FROM MEMORY
STORE CHR (15) TO INVERSE
STORE "                          " TO ZGBER
STORE F TO OK
§ 23,Ø SAY INVERSE + "ADRESSEN                    S T A R T " +;
 "                                  "
§ 6,21 SAY  "A D R E S S E N V E R W A L T U N G"
§ 8,21 SAY  "für Apple IIe   ergänzt am 1.5.1986"
§ 1Ø,21 SAY "Autor:                 Ernst Fischer"
§ 15,9 SAY "Letzter Zugriff :  " + DATE +;
 "              Anzahl Records : "
§ 15,68 SAY STR(R:NR,3)
DO WHILE .NOT. OK
  § 17,9 SAY "Bitte geben Sie das Datum ein     <TT/MM/JJ>;
             "GET DATE PICTURE '99/99/99'
  READ
  IF VAL($(DATE,1,2)) > 31;
    .OR. VAL($(DATE,1,2)) < 1;
    .OR. VAL($(DATE,4,2)) > 12;
    .OR. VAL($(DATE,4,2)) < 1;
    .OR. VAL($(DATE,7,2)) < 83
      ? CHR (7)
     ELSE
       STORE T TO OK
  ENDIF
ENDDO
ERASE
STORE $(DATE,1,2) + " " + $(DATE,4,2) + " " + $(DATE,7,2);
TO DATUM
SET DATE TO &DATUM
§ 23,Ø SAY INVERSE + "ADRESSEN                    S T A R T"+;
          "                    DATUM    :    " + DATE()
§ 17,5 SAY "Bitte Wasswort eingeben                    "
SET CONSOLE OFF
ACCEPT '       ' TO ZGBER
SET CONSOLE ON
ERASE
RELEASE OK, DATUM, DATE
IF ZGBER <> "PEEKER"
  RELEASE ALL
  ERASE
  QUIT
ENDIF
DO HMENUE
* Ende von AD-START
```

2. HMENUE.CMD

```
* DO HMENUE.CMD (Hauptmenueprogramm)
ERASE
DO WHILE T
§  1, Ø SAY "+=================================================;
=================================+"
§  2, Ø SAY "!                                                  ;
                                 !"
§  3, Ø SAY "!                                  Hauptmenue      ;
                                 !"
§  4, Ø SAY "+=================================================;
=================================+"
§  5, Ø SAY "!                                                  ;
                                 !"
§  6, Ø SAY "!       1.Eingabe der Personaldaten               ;
                                 !"
§  7, Ø SAY "!                                                  ;
                                 !"
§  8, Ø SAY "!       2.Verändern der Personaldaten             ;
                                 !"
§  9, Ø SAY "!                                                  ;
                                 !"
§ 1Ø, Ø SAY "!       3.Suchen nach Familienname                ;
                                 !"
§ 11, Ø SAY "!                                                  ;
                                 !"
§ 12, Ø SAY "!       4.Suchen nach Vornamen                    ;
                                 !"
§ 13, Ø SAY "!                                                  ;
                                 !"
§ 14, Ø SAY "!       5.Suchen nach Bemerkungen                 ;
                                 !"
§ 15, Ø SAY "!                                                  ;
                                 !"
§ 16, Ø SAY "!       6.Adressenaufkleber drucken               ;
                                 !"
§ 17, Ø SAY "!                                                  ;
                                 !"
§ 18, Ø SAY "!       7.Adressenlisten drucken / Adressen anschau;
en                               !"
§ 19, Ø SAY "!                                                  ;
                                 !"
§ 2Ø, Ø SAY "!       8.Löschen eines Datensatzes               ;
                                 !"
§ 21, Ø SAY "!                                                  ;
                                 !"
§ 22, Ø SAY "!       9.Ende                                    ;
                                 !"
§ 23, Ø SAY "+=================================================;
=================================+"
STORE F TO OK
 DO WHILE .NOT.OK
  SET CONSOLE OFF
  WAIT TO ANTW
  SET CONSOLE ON
```

```
    IF .NOT.ANTW$ '123456789'
       ? CHR(7)
        ELSE
         STORE T TO OK
    ENDIF
  ENDDO
 DO CASE
    CASE ANTW='1'
      DO AUFNAHME
    CASE ANTW='2'
      DO EDITFNAM
    CASE ANTW='3'
      DO SUCHFNAM
    CASE ANTW='4'
      DO SUCHVNAM
    CASE ANTW='5'
      DO SUCHBEME
    CASE ANTW='6'
      DO SCHREIBA
    CASE ANTW='7'
      DO SCHREIBL
    CASE ANTW='8'
      DO LOESCH
    CASE ANTW='9'
      RELEASE ALL
      ERASE
      USE B:ADRESSEN
      GO BOTTOM
      STORE # TO R:NR
      STORE DATE() TO DATE
      SAVE TO MEMORY
      CLEAR
      CANCEL
ENDCASE
ENDDO
* Ende von HMENUE.CMD
```

3. AUFNAHME.CMD

```
* DO AUFNAHME.CMD (Programm zur Aufnahme der Daten)
ERASE
SET TALK OFF
USE B:ADRESSEN
GO BOTTOM
STORE 'J' TO ANTW
DO WHILE !(ANTW) ='J'
   STORE ' ' TO ANTW
   APPEND BLANK
   REPLACE NAT WITH 'D'
   DO AUFMASKE
ENDDO
ERASE
§ 12,5 SAY 'Bitte keine Eingaben machen.Es wird eine Indexdatei;
 angelegt.'
INDEX ON FAM TO ALPHA
INDEX ON VOR TO BETA
INDEX ON VORE TO GAMA
CLEAR
RETURN
* Ende von DO AUFNAHME
```

4. AUFMASKE.CMD

```
* AUFMASKE.CMD (Maske zur Datenaufnahme)
ERASE
§  1, Ø SAY "+=================================================;
=============================+"
§  2, Ø SAY "!                                                 ;
                                                 !"
§  3, Ø SAY "!     Bitte geben Sie die Daten für das Adressregi;
ster ein...                                      !"
§  4, Ø SAY "+=================================================;
=============================+"
§  5, Ø SAY "!     Fam.Name"
§  5,14 GET fam
§  5,34 SAY "Vorname"
§  5,41 GET vor
§  5,56 SAY "Geb.Datum"
§  5,65 GET gebd
§  5,79 SAY "!"
§  6, Ø SAY "!                                                 ;
                                                 !"
§  7, Ø SAY "!     Ehefrau..................... Vorname"
§  7,41 GET vore
§  7,56 SAY "Geb.Datum"
§  7,65 GET gebde
§  7,79 SAY "!"
§  8, Ø SAY "!                                                 ;
                                                 !"
§  9, Ø SAY "!     PLZ"
§  9, 9 GET plz
§  9,16 SAY "Wohnort"
§  9,23 GET ort
§  9,4Ø SAY "Straße u.Nr."
§  9,52 GET str
§  9,79 SAY "!"
§ 1Ø, Ø SAY "!                                                 ;
                                                 !"
§ 11, Ø SAY "!     Land (Intern.KFZ-Zeichen als Abkürz.)"
§ 11,43 GET nat
§ 11,79 SAY "!"
§ 12, Ø SAY "+-------------------------------------------------;
----------------------------+"
§ 13, Ø SAY "!     Telefon Priv. Vorwahl"
§ 13,27 GET vwp
§ 13,35 SAY "Rufnr."
§ 13,41 GET tnrp
§ 13,49 SAY "Auslandsvorwahl"
§ 13,64 GET vwal
§ 13,79 SAY "!"
§ 14, Ø SAY "!                                                 ;
                                                 !"
§ 15, Ø SAY "!     Telefon Die.  Vorwahl"
§ 15,27 GET vwd
§ 15,35 SAY "Rufnr."
§ 15,41 GET tnrd
§ 15,79 SAY "!"
§ 16, Ø SAY "!                                                 ;
                                                 !"
§ 17, Ø SAY "!     Ehefrau... Telefon Dienstlich Vorwahl"
§ 17,43 GET vwde
§ 17,54 SAY "Rufnr."
§ 17,6Ø GET tnrde
§ 17,79 SAY "!"
§ 18, Ø SAY "+-------------------------------------------------;
----------------------------+"
§ 19, Ø SAY "!     Bemerkung"
§ 19,15 GET bem
§ 19,79 SAY "!"
§ 2Ø, Ø SAY "+-------------------------------------------------;
----------------------------+"
§ 21, Ø SAY "!     Weitere Eintragungen ? (J/N)"
§ 21,79 SAY "!"
§ 22, Ø SAY "+=================================================;
=============================+"
READ
RETURN
* Ende von DO AUFMASKE
```

5. AUSMASKE.CMD

```
* AUSMASKE.CMD (Gibt die gespeicherten Daten einer Person aus)
ERASE
§  1, Ø SAY "+=================================================;
=============================+"
§  2, Ø SAY "!     Daten der gesuchten Person...               ;
                                                 !"
§  3, Ø SAY "+=================================================;
=============================+"
§  4, Ø SAY "!     Fam.Name:"
§  4,15 SAY fam
§  4,35 SAY "Vorname:"
§  4,43 SAY vor
§  4,58 SAY "Geb.Datum:"
§  4,68 SAY gebd
§  4,79 SAY "!"
§  5, Ø SAY "!                                                 ;
                                                 !"
§  6, Ø SAY "!     Ehefrau..................... Vorname:"
§  6,43 SAY vore
§  6,58 SAY "Geb.Datum:"
§  6,68 SAY gebde
§  6,79 SAY "!"
§  7, Ø SAY "!                                                 ;
                                                 !"
§  8, Ø SAY "!     PLZ:"
§  8,1Ø SAY plz
§  8,17 SAY "Wohnort:"
§  8,25 SAY ort
§  8,42 SAY "Straße u.Nr.:"
§  8,55 SAY str
§  8,79 SAY "!"
§  9, Ø SAY "!                                                 ;
                                                 !"
```

```
§ 10, Ø SAY "!      Land (Intern.KFZ-Zeichen als Abkürz.):"
§ 10,44 SAY nat
§ 10,79 SAY "!"
§ 11, Ø SAY "+-------------------------------------------------------;
-----------------------------+"
§ 12, Ø SAY "!      Telefon Priv. Vorwahl:"
§ 12,28 SAY vwp
§ 12,36 SAY "Rufnr.:"
§ 12,43 SAY tnrp
§ 12,51 SAY "Auslandsvorwahl:"
§ 12,67 SAY vwal
§ 12,79 SAY "!"
§ 13, Ø SAY "!                                                        ;
                                        !"
§ 14, Ø SAY "!      Telefon Die.  Vorwahl:"
§ 14,28 SAY vwd
§ 14,36 SAY "Rufnr.:"
§ 14,43 SAY tnrd
§ 14,79 SAY "!"
§ 15, Ø SAY "!                                                        ;
                                        !"
§ 16, Ø SAY "!      Ehefrau... Telefon Dienstlich Vorwahl:"
§ 16,44 SAY vwde
§ 16,55 SAY "Rufnr.:"
§ 16,62 SAY tnrde
§ 16,79 SAY "!"
§ 17, Ø SAY "+-------------------------------------------------------;
-----------------------------+"
§ 18, Ø SAY "!      Bemerkung:"
§ 18,16 SAY bem
§ 18,79 SAY "!"
§ 19, Ø SAY "+-------------------------------------------------------;
-----------------------------+"
§ 2Ø, Ø SAY "!      Ist dies die gesuchte Person (J/N)"
§ 2Ø,4Ø GET GP
§ 2Ø,79 SAY "!"
§ 21, Ø SAY "+=======================================================;
=============================+"
READ
RETURN
* Ende von DO AUSMASKE.CMD
```

6. EDITFNAM.CMD

```
* EDITFNAM.CMD (VERÄNDERT DIE PERSONALDATEN)
SET TALK OFF
STORE 'J' TO ANTW
DO WHILE !(ANTW)='J'
   USE B:ADRESSEN INDEX ALPHA
   STORE ' ' TO ANTW
   ERASE
   § 4,26 SAY 'Verändern von Personaldaten'
   § 5,26 SAY '==========================='
   ?
   ?
   ACCEPT 'Familienname der gesuchten Person ' TO MFAM
   ?
   ACCEPT 'Vornamen der gesuchten Person ' TO MVOR
   FIND &MFAM
   IF # <> Ø
     STORE # TO R:NR
   ENDIF
   DO SUCH
   IF ENDE
     RETURN
   ENDIF
   USE B:ADRESSEN
   GO R:NR
   IF MVOR =' '
       LOCATE FOR FAM=MFAM
     ELSE
       LOCATE FOR FAM=MFAM.AND.VOR=MVOR
   ENDIF
   STORE FAM TO TFAM
   STORE VOR TO TVOR
   STORE VORE TO TVORE
   DO AUFMASKE
   STORE FAM TO T1FAM
   STORE VOR TO T1VOR
   STORE VORE TO T1VORE
   IF T1FAM <> TFAM
      ? ' Einen Moment bitte ...
      Aktualisierung der Fam-Indexdatei)'
      INDEX ON FAM TO ALPHA
   ENDIF
   IF T1VOR <> TVOR
      ? ' Einen Momen bitte ...
      (Aktualisierung der Vornam-Indexdatei)'
      INDEX ON VOR TO BETA
   ENDIF
   IF T1VORE <> TVORE
      ? ' Einen Moment bitte...
      (Aktualisierung der Vornam.Ehefr-Indexdatei)'
      INDEX ON VORE TO GAMMA
   ENDIF
ENDDO
RELEASE TFAM,T1FAM,TVOR,T1VOR,TVORE,T1VORE
RETURN
* Ende von EDITFNAM.CMD
```

7. SUCHFNAM.CMD

```
* SUCHFNAM.CMD (Sucht den Familiennamen)
USE B:ADRESSEN INDEX ALPHA
SET TALK OFF
ERASE
§ 4,1Ø SAY 'Datenausgabe für einen gewünschten Familiennamen'
§ 5,1Ø SAY '==================================================='
?
?
ACCEPT 'Familienname der gesuchten Person ' TO MFAM
FIND &MFAM
IF # <> Ø
   STORE # TO R:NR
ENDIF
DO SUCH
IF ENDE
  RETURN
ENDIF
USE B:ADRESSEN
GO R:NR
LOCATE FOR FAM=MFAM
STORE ' ' TO GP
DO AUSMASKE
DO WHILE !(GP)='N'
   STORE ' ' TO GP
   CONTINUE
   IF EOF
      ERASE
      § 12,Ø
      ? 'Weitere Personen, die ',MFAM,' heißen,gibt es nicht !'
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
      RETURN
   ENDIF
   DO AUSMASKE
ENDDO
RETURN
* Ende von DO SUCHFNAM.CMD
```

8. SUCHVNAM.CMD

```
* SUCHVNAM.CMD (Sucht den Vornamen)
USE B:ADRESSEN INDEX BETA
SET TALK OFF
   ERASE
   § 4,12 SAY 'Datenausgabe für einen gewünschten Vornamen'
   § 5,12 SAY '==========================================='
   ?
   ?
   ACCEPT 'Vorname der gesuchten Person ' TO MVOR
   FIND &MVOR
   IF #=Ø
      USE B:ADRESSEN INDEX GAMA
      FIND &MVOR
   ENDIF
     DO WHILE #=Ø
        ERASE
        § 12,Ø
        ?' Den Vornamen ',MVOR,' gibt es nicht !'
        ?
        ?' Bitte Eingabe wiederholen oder mit Return abbrechen !'
        ?
        ?
        ACCEPT ' Vorname der gesuchten Person ' TO MVOR
        IF MVOR =' '
           RETURN
        ENDIF
        USE B:ADRESSEN INDEX BETA
        FIND &MVOR
        IF #=Ø
           USE B:ADRESSEN INDEX GAMA
           FIND &MVOR
        ENDIF
     ENDDO
```

```
IF # <> Ø
   STORE # TO R:NR
ENDIF
USE B:ADRESSEN
GO R:NR
LOCATE FOR (VOR=MVOR.OR.VORE=MVOR)
STORE ' ' TO GP
DO AUSMASKE
DO WHILE !(GP)='N'
   STORE ' ' TO GP
   CONTINUE
   IF EOF
      ERASE
      § 12,Ø
      ? 'Weitere Personen, die ',MVOR,' heißen,gibt es nicht !'
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
      RETURN
   ENDIF
   DO AUSMASKE
ENDDO
RELEASE ALL
RETURN
* Ende von DO SUCHVNAM.CMD
```

9. SUCHBEME.CMD

```
* SUCHBEME.CMD (Sucht nach Bemerkungen)
USE B:ADRESSEN
SET TALK OFF
   ERASE
   § 4,8 SAY 'Datenausgabe für eine gewünschte Bemerkung'
   § 5,8 SAY '═══════════════════════════════════'
   ?
   ?
   ACCEPT 'Bemerkung, nach der gesucht wird ' TO MFAM
   LOCATE FOR MFAM$BEM
     DO WHILE EOF
        ERASE
        § 12,Ø
        ?' Die Bemerkung ',MFAM,' gibt es nicht !'
        ?
        ?' Bitte Eingabe wiederholen oder mit Return abrechen!'
        ?
        ?
        ACCEPT ' Bemerkung, nach der gesucht wird ' TO MFAM
        IF MFAM =' '
           RETURN
        ENDIF
        GO TOP
        LOCATE FOR MFAM$BEM
     ENDDO
STORE ' ' TO GP
DO AUSMASKE
DO WHILE !(GP)='N'
   STORE ' ' TO GP
   CONTINUE
   IF EOF
      ERASE
      § 12,Ø
      ? 'Weitere Bemerkungen mit ',MFAM,' gibt es nicht !'
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
      RETURN
   ENDIF
   DO AUSMASKE
ENDDO
RELEASE ALL
RETURN
* Ende von SUCHBEME.CMD
```

10. SCHREIBA.CMD

```
* SCHREIBA.CMD(Schreibt Adressenaufkleber)
USE B:ADRESSEN INDEX ALPHA
SET TALK OFF
   ERASE
   § 4,17 SAY 'Anfertigen von Adressenaufklebern'
   § 5,17 SAY '═══════════════════════════════'
   ?
   ?
   ACCEPT 'Familienname der gesuchten Person ' TO MFAM
   FIND &MFAM
   IF #<>Ø
      STORE # TO R:NR
   ENDIF
DO SUCH
IF ENDE
  RETURN
ENDIF
USE B:ADRESSEN
GO R:NR
LOCATE FOR FAM=MFAM
ERASE
STORE ' ' TO ANR
§ 4,35 SAY 'Anredemenü'
§ 5,35 SAY '═══════════'
§ 7,5 SAY '1.Keine Anrede '
§ 9,5 SAY '2.Keine Anrede bei Ehefrau'
§ 11,5 SAY '3.Anrede "Herrn"'
§ 13,5 SAY '4.Anrede "Frau" bei Verheirateten'
§ 15,5 SAY '5.Anrede "Herr+Frau"'
§ 17,5 SAY '6.Anrede "Familie"'
§ 19,5 SAY '7.Anrede "Frau" bei Alleinstehenden'
§ 21,5 SAY '8.Sonstige Anrede'
§ 23,Ø SAY 'Bitte wählen Sie '
STORE F TO OK
DO WHILE .NOT.OK
  SET CONSOLE OFF
  WAIT TO ANR
  SET CONSOLE ON
  IF .NOT.ANR$ '12345678'
    ? CHR(7)
    ELSE
     STORE T TO OK
   ENDIF
ENDDO
DO CASE
   CASE ANR='1'
      STORE ' ' TO ANREDE
   CASE ANR='2'
      STORE ' ' TO ANREDE
   CASE ANR='3'
      STORE 'Herrn' TO ANREDE
   CASE ANR='4' .OR. ANR='7'
      STORE 'Frau' TO ANREDE
   CASE ANR='5'
      STORE 'Herr u. Frau' TO ANREDE
   CASE ANR='6'
      STORE 'Fam.' TO ANREDE
   CASE ANR='8'
      ERASE
      § 12,Ø
      ACCEPT 'Welche Anrede wird gewünscht?' TO ANREDE
ENDCASE
STORE 'N' TO GP
DO WHILE !(GP)<>'J'
ERASE
   § 12,2Ø SAY ANREDE
   STORE ANREDE TO AN
 DO CASE
    CASE ANR='1'.OR.ANR='3'.OR.ANR='6'.OR.ANR='7'.OR.ANR='8'
    STORE (TRIM(VOR)+' '+FAM) TO NA
    § 13,2Ø SAY NA
    CASE ANR='2'.OR.ANR='4'
    STORE (TRIM(VORE)+' '+FAM) TO NA
    § 13,2Ø SAY NA
    CASE ANR='5'
    STORE (TRIM(VOR)+' + '+TRIM(VORE)+' '+FAM) TO NA
    § 13,2Ø SAY NA
ENDCASE
§ 14,2Ø SAY STR
IF NAT ='D'
    STORE (PLZ+' '+ORT) TO ORT1
   § 16,2Ø SAY ORT1
    ELSE
    STORE (NAT+' '+PLZ+' '+ORT) TO ORT1
   § 16,2Ø SAY ORT1
ENDIF
STORE ' ' TO GP
§ 2Ø,Ø SAY 'Ist dies die gesuchte Person? '
§ 22,Ø SAY 'J)a   W)eitersuchen  oder  R)eturn' GET GP
READ
DO CASE
   CASE !(GP)='W'
     CONTINUE
     IF EOF
        ERASE
        § 12,Ø
        ? 'Weitere Personen, die ',MFAM,' heißen,gibt es nicht !'
        SET CONSOLE OFF
        WAIT
        SET CONSOLE ON
        RETURN
     ENDIF
```

```
    CASE !(GP)='R'
      RETURN
    CASE !(GP)='J'
ENDCASE
ENDDO
ERASE
§ 5,Ø SAY 'Bitte den Drucker bereit machen und die ;
Adressaufkleber '
§ 7,Ø SAY 'einspannen!'
STORE '  ' TO ANZ
§ 1Ø,Ø SAY 'Anzahl der zu druckenden Adressaufkleber?';
 GET ANZ PICTURE '99'
READ
§ 15,2Ø SAY 'Folgende Justierung ist nötig:'
§ 16,2Ø SAY '═══════════════════════════'
§ 18,Ø SAY 'Linker Rand des Adressaufklebers auf Ø der Schiene'
§ 19,Ø SAY '                    ═══════════════ '
§ 21,Ø SAY 'Anlegeschiene muß mit Beginn zweites Loch ;
übereinstimmen.'
SET CONSOLE OFF
WAIT
SET PRINT ON
?
STORE Ø TO ZAHL
DO WHILE ZAHL < VAL(ANZ)
? CHR(27)+CHR(64)+CHR(27)+"3"+CHR(36)+CHR(27)+"G"+'          '+AN
* Drucker normieren.Zeilenabstand auf 36/216" und Doppeldruck
? '          '+NA
? '          '+STR
?
? CHR(27)+CHR(64)+CHR(27)+"E"+'          '+ORT1
* Drucker normieren und auf Fettschrift setzen
?
?
?
?
STORE ZAHL+1 TO ZAHL
ENDDO
? CHR(27)+CHR(64)+CHR(13)
* Drucker Normieren und Linefeed
SET PRINT OFF
SET CONSOLE ON
RELEASE ALL
RETURN
* Ende von DO SCHREIBA.CMD
```

11. SCHREIBL.CMD

```
* SCHREIBL.CMD (Druckt Adresslisten aus)
ERASE
SET TALK OFF
§ 3,19 SAY 'Adresslisten - und Etikettendruckprogramm'
§ 4,19 SAY '═══════════════════════════════════'
§ 1Ø,5 SAY '1.Adressen ansehen'
§ 12,5 SAY '2.Adressliste drucken'
?
STORE F TO OK
DO WHILE .NOT.OK
   SET CONSOLE OFF
   WAIT TO ANT
   SET CONSOLE ON
   IF .NOT.ANT$'12'
     ? CHR(7)
     ELSE
      STORE T TO OK
   ENDIF
ENDDO
USE B:ADRESSEN INDEX ALPHA
ERASE
IF ANT = '2'
   § 12,5 SAY 'Bitte den Drucker bereitmachen! Danach RETURN !'
   ?
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   SET PRINT ON
   ? CHR(27)+CHR(15)+CHR(27)+CHR(48)+CHR(27)+CHR(56)
* Drucker auf Schmalschrift, 1/8" Zeilenabst.+ Abschaltung der
* Papierenderkennung
ENDIF
STORE Ø TO LIN
DO WHILE .NOT.EOF
  DO CASE
    CASE ANT='2'
      DO WHILE (LIN < 42 .AND..NOT.EOF)
    CASE ANT='1'
       DO WHILE (LIN < 16 .AND..NOT.EOF)
  ENDCASE
      DO CASE
         CASE VORE <> '            '
            ? TRIM(VOR)+'+'+TRIM(VORE)+' '+FAM
            STORE LIN+1 TO LIN
         CASE VOR <> '            '
            ? TRIM(VOR)+' '+FAM
            STORE LIN+1 TO LIN
         CASE VOR = '            '
            ? FAM
            STORE LIN+1 TO LIN
       ENDCASE
      IF STR <> '                    '
        ? STR
        STORE LIN+1 TO LIN
      ENDIF
      IF ORT <> '                '
         ? TRIM(NAT)+' '+PLZ+' '+ORT
         ? '═══════════════════════'
         STORE LIN+2 TO LIN
      ENDIF
      IF VWAL<>'      '
         ? 'Tel.Nr.: '+TRIM(VWAL)+'-'+TRIM(VWP)+'-'+TNRP
         STORE LIN+1 TO LIN
       ELSE
         ? 'Tel.Nr.: '+TRIM(VWP)+'-'+TNRP
         STORE LIN+1 TO LIN
       ENDIF
     DO CASE
         CASE ((GEBD <> ' ' .AND. VWD <> ' ').AND.VORE<>'  ')
            ? TRIM(VOR)+' '+TRIM(GEBD)+' '+'Tel.Die.:'+VWD+TNRD
            STORE LIN+1 TO LIN
         CASE ((GEBD <> ' '.AND.VWD = '      ').AND.VORE <> ' ')
            ? TRIM(VOR)+' '+TRIM(GEBD)
            STORE LIN+1 TO LIN
         CASE (GEBD <> ' ' .AND.VWD <> ' ')
            ? TRIM(GEBD)+' Tel.Die.: '+VWD+TNRD
            STORE LIN+1 TO LIN
         CASE (GEBD = ' '.AND.VWD<> ' ')
            ? 'Tel.Die.: '+VWD+TNRD
            STORE LIN+1 TO LIN
      ENDCASE
      IF (GEBDE <> ' ' .OR. VWDE <> ' ')
         IF VWDE = '     '
            ? TRIM(VORE)+' '+TRIM(GEBDE)
            STORE LIN+1 TO LIN
            ELSE
            ? TRIM(VORE)+' '+TRIM(GEBDE)+' Tel.Die.:'+VWDE+TNRDE
            STORE LIN+1 TO LIN
         ENDIF
      ENDIF
      IF BEM<>' '
         ? 'Bemerk.: '+BEM
         STORE LIN+1 TO LIN
      ENDIF
      ?
      ?
      STORE LIN+2 TO LIN
      SKIP
         IF (LIN>35.AND.(ANT='2'))
            SET PRINT OFF
            ERASE
            § 12,5 SAY 'Bitte neues Blatt in den Drucker ;
einspannen.'
            § 14,5 SAY 'Danach mit RETURN weiterdrucken.'
            SET CONSOLE OFF
            WAIT
            SET CONSOLE ON
            SET PRINT ON
            STORE Ø TO LIN
         ENDIF
  ENDDO
 STORE Ø TO LIN
 IF ANT='1'
    STORE ' ' TO FRA
    § 23,Ø SAY 'A)bbrechen RETURN=Weiter' GET FRA
    READ
    ERASE
    IF !(FRA)='A'
      RETURN
    ENDIF
 ENDIF
DO CASE
   CASE ANT='1'
     ENDDO
   CASE ANT='2'
     ENDDO
ENDCASE
IF ANT='2'
   ? CHR(27)+CHR(64)
   * Drucker normieren
   SET PRINT OFF
ENDIF
RELEASE ALL
RETURN
* Ende von SCHREIBL.CMD
```

12. LOESCH.CMD

```
* LOESCH.CMD (Löscht die Personaldaten einer Person)
USE B:ADRESSEN INDEX ALPHA
SET TALK OFF
ERASE
§ 4,15 SAY 'Datenlöschung für eine bestimmte Person '
§ 5,15 SAY '═══════════════════════════════'
?
?
ACCEPT 'Familienname der gesuchten Person ' TO MFAM
FIND &MFAM
IF # <> Ø
   STORE # TO R:NR
ENDIF
DO SUCH
IF ENDE
  RETURN
ENDIF
USE B:ADRESSEN
GO R:NR
LOCATE FOR FAM=MFAM
STORE ' ' TO GP
DO AUSMASKE
DO WHILE !(GP)='N'
   STORE ' ' TO GP
   CONTINUE
   IF EOF
      ERASE
      § 12,Ø
      ? 'Weitere Personen, die ',MFAM,' heißen,gibt es nicht !'
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
      RETURN
   ENDIF
   DO AUSMASKE
ENDDO
ERASE
§ 5,Ø SAY 'A C H T U N G,  A C H T U N G,  A C H T U N G,  ;
A C H T U N G'
§ 6,Ø SAY '═══════════════════════════════════════;
═══════'
§ 9,Ø SAY 'Folgende Person wird gelöscht:'
§ 1Ø,Ø SAY '═══════════════════════════'
§ 12,Ø SAY 'Familienname :'+FAM
§ 14,Ø SAY 'Vorname      :'+VOR
§ 16,Ø SAY 'Straße       :'+STR
§ 18,Ø SAY 'Ort          :'+PLZ+' '+ORT
STORE ' ' TO LOESCH
§ 22,Ø SAY 'Ist dies die gewünschte Person zum Löschen? ;
(J/N)' GET LOESCH
READ
IF !(LOESCH)='J'
   DELETE
   PACK
   ERASE
   § 12,Ø SAY 'Es wurden die Daten der Person '+MFAM+';
 gelöscht!!!'
   § 14,Ø SAY 'Bitte keine Eingaben machen; es werden;
 die Indexdateien'
   § 15,Ø SAY 'aktualisiert.'
   INDEX ON FAM TO ALPHA
   INDEX ON VOR TO BETA
   INDEX ON VORE TO GAMA
ENDIF
RELEASE ALL
RETURN
* Ende von DO LOESCH.CMD
```

13. SUCH.CMD

```
* SUCH.CMD (Sucht nach gegebenen Kriterien)
STORE F TO ENDE
DO WHILE #=Ø
   ERASE
   § 12,Ø
   ?' Den Familiennamen ',MFAM,' gibt es nicht !'
   ?
   ?' Bitte Eingabe wiederholen  oder  mit Return abbrechen !'
   ?
   ?
   ACCEPT ' Familienname der gesuchten Person ' TO MFAM
   IF MFAM =' '
      STORE T TO ENDE
      RETURN
   ENDIF
   FIND &MFAM
   STORE # TO R:NR
ENDDO
RETURN
* Ende von SUCH.CMD
```


Hüthig FACHBUCHTIP:

Applesoft BASIC

Tips und Tricks

von Frank Bühler

1985, 241 S., 40 Abb., kart., DM 38,—
ISBN 3-7785-1094-0

Begleitdiskette DM 44,—
ISBN 3-7785-1291-9

Das Buch enthält eine komplette Beschreibung aller möglichen Applesoft-Befehle und zeigt an einem Beispiel die erforderliche Syntax auf. Ausgearbeitete Unterroutinen können leicht in eigene Programme übernommen werden.

Das Buch zum Apple-Writer II/IIe

von Hans Gabriel

1986, 157 S., kart., DM 35,—
ISBN 3-7785-1234-X

Begleitdiskette DM 44,—
ISBN 3-7785-1337-0

„Das Buch zum Apple-Writer II/IIe" wendet sich an alle, die dieses Textverarbeitungssystem schon einsetzen oder noch einsetzen wollen.

Apple IIc

Handbuch für Anwender und Programmierer

von Carl-Ulrich Wassermann

1985, 324 S., zahlr. Abb., kart., DM 35,—
ISBN 3-7785-1157-2

Wenn Sie die Leistungsfähigkeit Ihres Apple IIc bisher noch nicht ausschöpfen konnten, brauchen Sie dieses Buch.

DTACK-68000-Board

Ein Erfahrungsbericht

von Wolfgang Trier

Wir alle kennen das Problem nur zu gut: Für viele Aufgaben ist die alte 6502-CPU des Apple II einfach zu langsam, woran auch der Einsatz einer Z80-Karte nichts Entscheidendes ändert. Also was tun? Ein neues System scheidet meist aus zwei Gründen aus: Die Menge der geschriebenen und gesammelten Software verbieten einen Systemwechsel, und außerdem trennt man sich von einer bekannten Maschine nur sehr ungern, da dies mit großem Zeit- und Kostenaufwand verbunden ist.

Um diesen Mißstand zu beheben, gibt es eine Erweiterung, die auf der Basis des MC 68000 von Motorola arbeitet, der neben seiner internen 32-Bit-Struktur auch noch mit der 12,5fachen Taktfrequenz läuft: Das DTACK-Board.
Es handelt sich hierbei um ein externes Board, das über ein Interface in einem beliebigen Slot mit dem Apple verbunden wird. Dessen mit 12,5 MHz getakteter Prozessor greift ohne „wait states" auf statisches RAM zu. (Anm.d.Red.: Zum Vergleich arbeitet der Macintosh wegen der Wartezyklen mit ca. 5,6 MHz.)
Die Karte ist sehr sauber mit Präzisionssockeln aufgebaut und macht einen recht aufgeräumten Eindruck. Die Ein- und Ausgabe erfolgt wahlweise über den Apple (Interface-Karten) oder ist auch direkt vom Expansion-Bus des DTACK-Boards möglich. Der DMA-Kanal des Apple wird nicht blockiert, so daß Harddisk-Controller usw. weiterhin unverändert verwendbar sind. Auch eine Floating-Point-Prozessorkarte mit dem NS 16081 ist erhältlich und wird von der Software unterstützt.
Der 68000 und der 6502 des Apple arbeiten parallel (gleichzeitig). Dadurch läßt sich der als intelligentes I/O genutzte Apple z.B. als Printer-Spooler einsetzen, ohne daß der 68000 durch die Printer-Ansteuerung blockiert wird.
Erfreulicherweise ist auch die Software-Unterstützung hervorragend, so daß alle Pascal-, Fortran- und Modula-Programme in UCSD-P-Code ohne neue Übersetzung unverändert laufen. Ferner ist das von Digital Research entwickelte und weit verbreitete CP/M 68K, das auch einen C-Compiler enthält, lauffähig.
Darüber hinaus ist ein von einer deutschen Firma entwickelter und in 68000-Code geschriebener und daher sehr schneller 68000/68020-Assembler erhältlich.

1. Das UCSD-System

1.1. P-Code-Interpreter INTER86

Das gesamte UCSD-Betriebssystem ist durch einen P-Code-Interpreter in 68000-Code mit Namen INTER68 voll verfügbar. Als besonders angenehm, vor allem für die immer zahlreicher werdenden Pascal-Freunde, erweist sich die Tatsache, daß mit INTER68 nicht nur eine enorme Geschwindigkeitsverbesserung der alten Programme möglich ist, sondern viele zusätzliche Features geboten werden, die man bisher schmerzlich vermißt hat.
INTER68 interpretiert P-Code, wie er im „Apple Pascal Operating System Reference Manual" definiert ist und ist daher aufwärtskompatibel zum Apple-Pascal. So laufen z.B. alle von Apple mit dem UCSD-System gelieferten Demo-Programme ohne erneute Übersetzung oder Modifizierung sofort, nur eben sehr viel schneller (ebenso die Turtlegraphics-Routinen).
INTER68 erlaubt Compiler-Raten von über 2000 Zeilen pro Minute, was gegenüber den bekannten 300 Zeilen pro Minute auf dem Apple allein schon einen riesigen Fortschritt darstellt. Wo man früher bei langen Programmen minutenlang warten mußte, nur um festzustellen, daß kurz vor dem Programmende noch ein Syntax-Fehler saß, macht sich das bei mir heute kaum negativ bemerkbar, da die Dots des Compilers ohnehin schneller sind als man zählen kann, so daß sich kaum ein nennenswerter Zeitverlust ergibt. Das Programmieren ist jetzt weit weniger ermüdend und macht viel mehr Spaß. Außerdem sind die Programme typischerweise zwischen 12- und 20mal schneller als bisher, je nach Grad des verwendeten I/O (z.B. Diskzugriff etc., der natürlich nicht beschleunigt werden kann).

INTER68 liegt ab Adresse $10.000 aufwärts und bietet einen 60K-Arbeitsspeicher; die unteren 4K können nicht benutzt werden, da sie vom Bootstrap-ROM benötigt werden. Vor allem die Systemprogramme profitieren ganz besonders von diesem zusätzlichen Platz. Programme, die bisher auf die Compiler-Swapping-Option angewiesen waren, können jetzt häufig ohne zeitaufwendiges Disk-Swapping compiliert werden.
Der Editor kann bis zu 60 (!) Blocks im Arbeitsspeicher halten (oder 64, wenn System Swapping eingeschaltet ist).
INTER68 läßt mit dem CONFIGURE-Programm eine Aufteilung des Speichers in Arbeitsspeicher und RAM-Disk zu, deren Volume-Nummer frei definierbar ist. Es ist auch möglich, mit CONFIGURE die RAM-Disk als Boot-(Root-)Volume zu installieren.
Je nach Speicherausbau greift das System kaum noch auf die Diskette zu, was neben der Ruhe auch eine enorme Geschwindigkeitsverbesserung im Sprung zwischen den verschiedenen Systemebenen oder -programmen bedeutet.

INTER68 benutzt das BIOS des Apple-Pascal, um mit der Peripherie zu kommunizieren (liegt in der 16K-Karte). Solange die bekannten Aufruf-Konventionen befolgt werden, kann sich z.B. jeder Anwender auch ein eigenes Host-Interface schreiben, da man am mitgelieferten Sourcecode des Standard-Software-Interfaces genau sehen kann, wie die Kommunikation zu erfolgen hat. Vier spezielle BIOS-Aufrufe (UserRead, UserWrite, UserStatus, UserUnit) wurden aufgestellt, um die Unit-Nummern von 128 bis 255 zu nutzen. So kann z.B. mit anwenderspezifischer Peripherie (A/D-Wandler, 640K-Drive usw.) über die Standard-Pascal-Funktionen BLOCKREAD/BLOCKWRITE komfortabel kommuniziert werden.
INTER68 kann Code von positivem und negativem Byte-Geschlecht interpretieren (Apple Pascal kann nur negatives Byte-Geschlecht interpretieren); daher können die Programme Segmente beider Byte-Geschlechter beinhalten, was die Verarbeitungsgeschwindigkeit nicht beeinflußt.

1.2. Lieferumfang

Geliefert werden auf einer Diskette folgende Files:
SYSTEM.68000 – enthält INTER68 im Phase-Zero-Format.
SYSTEM.COMPILER – ein modifizierter

Compiler, der automatisch das Byte-Flippen berücksichtigt und Code generiert, für den die transzendenten Funktionen, die schon im INTER68 enthalten sind, zugänglich sind.
SYSTEM.LIBRARY – enthält Longinteger, Turtlegraphics und Applestuff, die in 68000-Code geschrieben sind.
MAKEPASCAL – Dieses Utility-Programm überführt eine Kopie des SYSTEM.PASCAL in einen File 68000.PASCAL. MAKEPASCAL patcht das Operating-System an sechs Stellen, wie man am mitgelieferten Textfile MAKEPASCAL.TEXT leicht ersehen kann.
UPLOAD – transferiert INTER68 auf das DTACK-Board. Wen interessiert, wie das geschieht, schaut sich am besten die Files UPLOAD.PAS.TEXT und UPLOAD.ASM.TEXT an.
CALL68 – startet INTER68 im 68000 und die Kommando-Prüfroutinen im Software-Interface (apple-seitig). Auch wird hier die höchstmögliche RAM-Adresse auf dem DTACK-Board gesetzt, um die Möglichkeit einer einfachen Parameteränderung zu gewähren. Alle Interface-Routinen liegen auch als Source-Code in den Files CALL 68.AS1.TEXT und CALL68.AS2.TEXT vor, die in ihrer Bedeutung im Handbuch ausführlich beschrieben sind.
UPLOAD und CALL68 sind getrennt, da man normalerweise nach einem Rücksprung in das Apple Pascal den im DTACK-Board liegenden Interpreter nicht neu laden muß.
FLIP – flippt 16-Bit-Einheiten im Block 0 eines jeden spezifizierten Files. Wenn FLIP zweimal angewendet wird, ist der ursprüngliche Zustand wieder hergestellt. Man muß z.B. beim Linken von Files unterschiedlichen Byte-Geschlechtes flippen.

1.3. Neuer Datentyp Double

Besonderer Erwähnung bedarf weiterhin der neue Datentyp Double, der doppelt genaue Gleitkomma-Arithmetik mit 15 (!) signifikanten Stellen ermöglicht. Die 1000fache Bearbeitung der Zeile

a := tan (arctan (exp (ln (sqrt (sqr (a))))))/a - 1;

ergab einen Fehler von nur 3.939 E-17, im Gegensatz zu 2.269 E-7 des Apple Pascal, was eine Genauigkeitssteigerung um den Faktor 10.000.000.000 darstellt.
Wer will, kann für DM 195.- einen Native Code Generator (NCG68) erstehen, der vorhandene Programme noch einmal um bis zu 8mal schneller laufen läßt. Man kann sogar wählen, ob man die Optimierung des gesamten Programmes oder nur eines Segmentes wünscht.

1.4 Macro-Assembler ASM68020

Ein weiterer wesentlicher Bestandteil des Pascal-Systems ist ein hervorragender Macro-Assembler.
Mit **ASM68020** erhält man einen in 68000 geschriebenen Macro-Assembler für den 68000, 68010 und den 68020 mit „Conditional Assembly" und lokalen Labels. Die Codefiles des ASM68020 können an alle Pascal- oder Fortran-Programme „gelinkt" werden oder aber auch als selbständige Programme ausgeführt werden.
Es handelt sich dabei um einen „one pass"-Assembler und ist damit wesentlich schneller als zweistufige Assembler. Diesen Vorteil muß man jedoch mit einer längeren Symbol Table und kleinen Einschränkungen im Gebrauch von Labels bezahlen.

Wenn ASM68020 auf einer RAM-Disk vorliegt, sind Zeiten von über 3000 Zeilen pro Minute bei einem 25.000-Zeilen-Programm möglich, ansonsten liegt der Durchsatz immerhin noch bei etwas über 1500 Zeilen pro Minute.
Der Gebrauch des ASM68020 entspricht etwa dem, was ich vom Apple-Pascal-6502-Assembler gewohnt war. Der File SYSTEM.ASSEMBLER enthält sowohl den Assembler als auch die Opcodes. Wenn 68020.ERRORS verfügbar ist, gibt der Assembler nicht nur die Nummer, sondern auch den Fehler genau an.
Der gleiche Assembler steht auch unter CP/M 68K zur Verfügung.

2. Das CP/M-68K-System

Eine Diskette (CPM68) ermöglicht die Installierung des CP/M-68K-Betriebssystems.
Mit diesem CP/M von Digital Research ist es möglich, auf dem DTACK-Board Programme in den Sprachen Pascal, Modula, Fortran, C (wird mitgeliefert) und Forth zu schreiben. Auch der aus dem P-Code-System von INTER68 bekannte schnelle Assembler ASM 68 läuft unter CP/M weit schneller als der von Digital Research mitgelieferte 68000-Assembler.

2.1. Lieferumfang

Auf der CPM68-Diskette befinden sich folgende Files:
CP/M.SYS – Objekt-File des BIOS für das Apple-68000-System.
UPLOAD.COM – Transferiert CP/M.SYS und CP/M.REL (relozierbare Version des CP/M 68K von Digital Research) zum 68000-Board.
CALL68.COM – Startet sowohl das 68K-Operating-System als auch apple-seitig die „Command Fetch Loop".
CONFIG.COM – Paßt das CP/M-Operating-System den individuellen Wünschen an.

Mit dem CONFIG-Programm kann jeder das System auf seine eigenen Bedürfnisse zuschneiden. Das Programm ändert folgende Parameter:

– Länge der TPA (Default: 96K = $00F000)
– Höchste RAM-Adresse (Default: 96K = $018000)
– Adresse der USER-INPUT-Routine (Default: $000000)
– Adresse der USER-OUTPUT-Routine (Default: $000000)
– Slot-Nummer des 68000-Interfaces (Default: 2)

Unter CP/M 68K können zusätzlich 8 verschiedene anwenderspezifische Ein- und Ausgaberoutinen (USER-INPUT/USER-OUTPUT) installiert werden. Die Anfangsadressen der USER-INPUT-/USER-Output-Routinen werden mit CONFIG definiert.

Alle Devices (z.B. Harddisks, 640K-Drives usw.), die unter Z80-CP/M verfügbar sind, werden durch CALL68 automatisch erkannt und stehen dem Benutzer unter CP/M 68K ohne zusätzliche Installierung unverändert zur Verfügung.
Beide Betriebssysteme, Z80-CP/M und CP/M 68K, residieren im RAM-Bereich des Systems, was die Ausnutzung der Vorteile des jeweiligen Systems unproblematisch macht.
Dies ermöglicht die simultane Benutzung von bestehenden Z80-Programmen (z.B. Wordstar, dBase etc.) und 68000-Programmen. Darüber hinaus können auf der gleichen Diskette Z80-CP/M- und CP/M-68K-Programme gemischt gespeichert werden, da beide Betriebssysteme das gleiche Diskettenformat verwenden. Durch das Suffix „.68K" und „.COM" wird die Zugehörigkeit zum jeweiligen Betriebssystem automatisch erkannt und macht so einen Systemabsturz durch Verwechslung unmöglich.

Je nach Speicherangebot wird automatisch eine Pseudo-Disk angelegt, die den Platz zwischen dem Ende der TPA und der höchsten RAM-Adresse nutzt.
Die Bezeichnung der Pseudo-Disk ist zwischen „A:" und „P:" frei wählbar.
Diese logische Einheit kann in der gleichen Weise wie eine externe Floppy genutzt werden, nur daß die Geschwindigkeit des Datentransfers wesentlich steigt. Man kann auf P: Files editieren, assemblieren, compilieren, alles natürlich schneller als mit einer Harddisk.

Dank der sehr ausführlichen Dokumentation ist es auch hier ohne weiteres möglich, eigene Treiber zu schreiben und eigene Karten (wie z.B. A/D-D/A-Wandler, Graphic Boards usw.) unter CP/M einzubinden. Es besteht ebenfalls die Möglichkeit, Daten direkt über das DTACK-Board ein- und auszugeben.
Leider ist in der Bundesrepublik wegen der amerikanischen Ausfuhrbestimmungen das Original-DTACK-Board der Firma Digital Acoustics, Santa Ana, Ca. nicht zu beziehen. Das Ingenieurbüro Fuchs in Aachen stellt jedoch eine verbesserte Version her, so daß man auch bei uns nicht auf den enormen Geschwindigkeitsvorteil eines am Apple betreibbaren 68000 verzichten muß.

Plusworks

Appleworks auf dem Apple II+

Ein Erfahrungsbericht von Herwig Cuypers

Peeker-Leser und Tester Ingenieur H. Cuypers ist Belgier, der seit seiner Schulzeit keinen längeren deutschen Text mehr geschrieben hat. Trotzdem ist ihm der deutschsprachige Bericht, den wir natürlich etwas überarbeiten mußten, hervorragend gelungen. Versetzen Sie sich in die Lage, Sie müßten einen Apple-Bericht in Französisch schreiben, und Sie werden sich über die sprachliche Leistung von Herrn Cuypers nur wundern können. us

1. Appleworks und Apple II+

Das integrierte Softwarepaket Appleworks wurde bekanntlich nur für die neueren Apple-Typen IIe und IIc entwickelt. Da Appleworks verstärkt vom RAM der erweiterten 80-Zeichenkarte Gebrauch macht, ist es grundsätzlich auf dem Apple II+ nicht lauffähig. Auch eine 128K-RAM-Karte bringt keine Abhilfe, da diese nicht in den gesamten Speicherbereich des Motherboards gemappt ist. Eine eigene Version für den Apple II+ wurde von Apple, wohl auch in Hinblick auf die Flut der kompatiblen Rechner, nicht herausgebracht.
Nun wird jedoch seit einiger Zeit in den USA von der Firma Norwich Data Services Ltd. das Patchprogramm Plusworks angeboten. Dieses ist in den drei Versionen *Plusworks*, *Plusworks XM* und *Plusworks XMP* erhältlich.

2. Plusworks

Plusworks patcht die Appleworks-Boot-Diskette. An Hardware benötigt es einen Apple II+ mit Applesoft-BASIC im ROM, 64K RAM, eine Tastatur mit Groß- und Kleinschreibung (Shift-Key-Modifikation wird unterstützt) und eine 80-Zeichenkarte. Angeboten wird es für ca. $23 inkl. Porto und Verpackung.

Plusworks XM stellt die gleichen Anforderungen an die Hardware, bietet auch die gleichen Möglichkeiten, unterstützt jedoch zusätzlich eine Reihe von RAM-Karten bis 1024K. Plusworks XMP schließlich unterstützt noch zusätzlich die in den USA sehr verbreiteten PCPI- und ALS-CP/M-Karten mit jeweils 64K RAM. Die XM- und XMP-Versionen können auch auf einem Apple IIe sinnvoll eingesetzt werden, wenn man etwa den Desktop-Bereich mit einer RAM-Karte erweitern oder statt der erweiterten 80-Zeichenkarte lieber seine alte Videoterm verwenden möchte. Beide Programme sind für jeweils $53 inkl. Porto und Verpackung erhältlich.

3. Plusworks XM

Ich selbst bin (immer noch) Besitzer eines Apple II+, ausgerüstet mit einer Videx-Videoterm und einer Saturn 128K-Karte. Ich bestellte daher in den USA Plusworks XM. Ungefähr 2 Wochen nach der Bestellung traf es gut verpackt per Post bei mir ein. Das Programm wird auf einer einseitigen, kopierbaren 5 1/4-Zoll-Diskette geliefert. Die Dokumentation beschränkt sich auf eine zehnseitige Broschüre in englischer Sprache.

4. 80-Zeichenkarten

Plusworks XM unterstützt folgende 80-Zeichenkarten:
- Videx Videoterm (mit inversem Zweitzeichensatz)
- Ultraterm
- Viewmax 80
- Franklin
- Viewmaster 80
- Magnum 80
- Wizard 80
- Smarterm II

Die 80-Zeichenkarte muß wie üblich in Slot 3 installiert sein.

5. RAM-Karten

In der mir vorliegenden Version unterstützt Plusworks XM alle RAM-Karten der Firmen Legend Industries, Titan/Saturn, Abacus und Prometheus sowie die dazu kompatiblen. Da der Apple II+ über 64K RAM verfügen muß, ist außerdem eine 16K-Karte in Slot 0 erforderlich. Hat die RAM-Karte mehr als 128K, kann der über 128K hinausgehende Bereich zusätzlich als RAM-Disk verwendet werden.

6. Installation

Der gesamte Installierungsvorgang findet im Laufwerk 1 statt. Dazu wird die Plusworks-XM-Diskette eingelegt und gebootet. Nach dem Booten fordert Plusworks zum Einlegen der Appleworks-Startup-Diskette auf. Ist dies geschehen, fordert Plusworks wenig später zum erneuten Einlegen der Plusworks-Diskette auf. Kurz danach erscheint ein Menü mit einer Auswahl von verschiedenen 80-Zeichenkarten. Nach Eingabe der verwendeten 80-Zeichenkarte schaltet Plusworks auf 80-Zeichendarstellung um und testet den 80-Z/Z-Bildschirm. Zum letzten Mal fordert Plusworks dann zum Einlegen der Appleworks Startup-Diskette (oder besser einer Kopie) auf. Die Plusworks-Diskette benötigt man nun nicht mehr.
Das Patchprogramm bittet nun um Antwort auf verschiedene Fragen. Zuerst kommt die Frage nach der Art der Tastatur. Man hat die Wahl zwischen einer ASCII-Tastatur und einer Apple-Tastatur mit Shift-Key-Modifikation. Die nächste Frage bittet um Angabe einer Ersatztaste für die beim Apple II+ fehlende Delete-Taste. Plusworks empfiehlt Control-Q. Als nächstes erfragt Plusworks eine Ersatztaste für die offene Apfel-Taste. Plusworks schlägt hier Control-O vor. Ersatzweise kann aber auch die Drucktaste 1 des Joysticks verwendet werden, eine Möglichkeit, die mir etwas umständlich erscheint.
Nun fragt Plusworks, ob ein Druckerpatch notwendig ist, warnt jedoch gleichzeitig davor, vorschnell mit „Y" zu antworten. Es empfiehlt statt dessen, zuerst Appleworks so auszuprobieren, um festzustellen, ob der Patch überhaupt vonnöten ist.
Der Rest des Konfigurationsvorgangs läuft weitgehend automatisch ab, Plusworks XM fragt nur noch nach dem Slot der RAM-Karte. Diese wird automatisch identifiziert. Verfügt sie über mehr als 128K, fragt Plusworks noch, ob der verbleibende Speicherbereich als RAM-Disk verwendet werden soll.
Appleworks ist jetzt auf dem Apple II+ lauffähig. Wird irgendeine Taste gedrückt, bootet die Diskette.

7. Die Bedienung des modifizierten Appleworks

Die geänderten Tastenfunktionen sind gut gewählt und erlauben ein bequemes Arbeiten mit Appleworks.

7.1. Tastatur allgemein

Alle „Offener-Apfel"-Befehle von Appleworks können mit der gewählten Ersatztaste simuliert werden. Will der Benutzer z.B. eine Hardcopy des Bildschirminhalts ausdrucken, tippt er zuerst auf „Control-O", dann auf „H". Nachdem die Ersatz-Apfeltaste gedrückt wurde, erscheint an der aktuellen Cursorposition ein inverses „+". Um dies nach versehentlichem Drücken von Control-O wieder zum Verschwinden zu bringen, muß man einen ungültigen „Apfelcode" eingeben, z.B. Control-O - 0.
Zieht der Anwender es jedoch vor, die Joystick-Taste 1 als Apfeltaste zu verwenden, muß diese *gleichzeitig* mit dem betreffenden Codezeichen betätigt werden.

Auf dem Apple II+ gibt es bekanntlich nur zwei Cursortasten. Plusworks XM sieht als Ersatz für einen Cursorblock den z.B. vom Wordstar bekannten „ESDX-Cluster" vor. Die vorhandenen Pfeiltasten können jedoch ebenso wie der „ESDX-Cluster" verwendet werden. In diesem Cluster entsprechen:
Control-E = Cursor nach oben
Control-D = Cursor nach rechts
Control-X = Cursor nach unten
Control-S = Cursor nach links
Auf der Tastatur sieht das wie folgt aus:

```
  E
S   D
  X
```

Abhängig von der Version der Tastatur werden die Cursorbewegungen, bei denen man im ungepatchten Appleworks die „Offener-Apfel"-Taste betätigt, unterschiedlich ersetzt.

7.2 Original Apple-II+-Tastatur

Hier wird die „Esc"-Taste als Lead-in-Taste verwendet. Will man jedoch ein echtes „Escape" erzeugen, muß man zweimal die „Esc"-Taste betätigen. Die „Apfel-Cursorbewegungen" (eine Seite nach oben/unten, ein Wort rechts/links) werden durch *gleichzeitiges* Betätigen der Control- und Shifttaste mit E, S, D oder X erzeugt. Delete erhält man mit Shift-Linkspfeil, Control-A fungiert als Caps-Lock-Taste. Eine Reihe von Sonderzeichen kann man erzeugen, indem man erst die Escape-Taste und danach ein bestimmtes Zeichen tippt. „Tab" erzeugt man mit Control-I, den umgekehrten „Tab" mit Shift-Control-I.

7.3. ASCII-Tastatur

Hier ist die Sache etwas einfacher. Der „ESDX"-Cluster wurde einfach um folgende Codes erweitert:
Control-A = ein Wort nach links
Control-F = ein Wort nach rechts
Control-R = eine Seite nach oben
Control-C = eine Seite nach unten
Control-V = umgekehrter „TAB"-Befehl
Auf der Tastatur sind diese Tasten etwa so angeordnet:

```
    E   R
A   S   D   F
    X   C   V
```

Diese Anordnung prägt sich meines Erachtens sehr leicht ein und ermöglicht ein bequemes Arbeiten.

8. RAM-Disks

Da Appleworks ein ziemlich großes Programm ist, können nicht alle Module auf einmal in den Hauptspeicher geladen werden. Daraus resultiert ein relativ häufiger Zugriff auf das Laufwerk 1, was natürlich Zeit kostet. Um den Zugriff auf das Hauptlaufwerk zu minimieren, verfügt Appleworks über eine dynamische Speicherverwaltung. Beim Startup-Prozeß werden soviele Module wie möglich in den freien Desktop-Speicher geladen. Ist der Desktop-Bereich groß genug, braucht jedes Modul nur einmal geladen zu werden. Da jedoch die Benutzerdaten höchste Priorität besitzen, verringert sich der freie Desktopbereich um so mehr, je größer die Datenmenge wird. So werden zunehmend Module aus dem freien Desktopspeicher gelöscht, und Appleworks muß wieder häufiger auf das Laufwerk 1 zugreifen. Werden nun RAM-Karten mit einer Speicherkapazität von 256K und mehr verwendet, bietet Plusworks zwei verschiedene Arten der RAM-Organisation an.

8.1. Dynamische RAM-Disk

Wird diese Option gewählt, lädt Appleworks beim Startup-Prozeß alle Programm-Module in den freien Desktop-Bereich. Die Startup-Prozedur wird daduch recht langwierig, danach aber greift Appleworks nur noch sehr selten auf die Boot-Diskette zu. Diese Funktion steht bei größeren RAM-Karten automatisch zur Verfügung, muß jedoch beim Startup-Prozeß explizit aufgerufen werden.
Von Vorteil ist hierbei der große Desktop-Bereich, doch steht dem die lange Bootdauer nachteilig gegenüber. Bootet man jedoch Appleworks einmal und arbeitet dann sehr lange Zeit damit, ist diese Möglichkeit durchaus sinnvoll einzusetzen.

8.2. Feste RAM-Disk

Hier wird die RAM-Karte als echte RAM-Disk verwendet. Der Speicher wird in festen Desktop-Bereich und Pseudo-Laufwerk unterteilt. Von Nachteil ist der Verlust von 136K bis 144K Desktop-Speicher. Demgegenüber dauert der gesamte Startup-Prozeß nur ca. 20 Sekunden (!) und man kann die Appleworks-Diskette aus dem Laufwerk herausnehmen und eine Datendiskette einlegen.

9. Druckerpatch

Dieser Patch setzt bei jedem gesendeten Zeichen das 8. Bit. Einige Druckerschnittstellen scheinen diesen Patch zu benötigen, die meisten jedoch nicht. Plusworks XM empfiehlt diesen Patch nur, wenn bei der Hardcopy oder der Ausgabe von Daten Probleme auftreten. Jedenfalls brauchten weder mein Epson FX-80 noch mein Mannesmann-Tally MT-80 diesen Patch. Das Handbuch zu Plusworks nennt eine Reihe von Symptomen, bei denen der Druckerpatch angebracht ist.

10. Fazit

Für ca. $53 erhält man ein Patchprogramm, das Appleworks wirklich uneingeschränkt auf dem Apple II+ lauffähig macht. Allerdings ist eine RAM-Karte mit mindestens 128K dringend erforderlich. Einen Desktop-Bereich von 10K kann ich jedenfalls nur noch als lächerlich bezeichnen. Mit einer 128K-Karte beträgt er dann immerhin 136K und ist damit genauso groß wie eine ProDOS-Datendiskette. Die mitgelieferte Broschüre enthält alle notwendigen Informationen; das amerikanische Englisch bereitet jedoch vielleicht manchen Anwendern Probleme.
Ich arbeite nun schon seit einigen Monaten mit dem gepatchten Appleworks auf einem Apple II+ und einem -Kompatiblen mit Voll-ASCII-Tastatur. Das angepaßte Appleworks erscheint mir auf dem kompatiblen Rechner wegen der Tastatur ein wenig benutzerfreundlicher.
Nur ein einziges Mal trat ein kleines Problem auf. Nachdem ich aus dem Apple II+ irgendeine Karte entfernt hatte, lief mit Appleworks überhaupt nichts mehr. Es bootete zwar, doch konnte das Datum nicht geändert werden, und im Hauptmenü wurde jede Eingabe mit einem fröhlichen Piepston quittiert. Nach sehr langer Fehlersuche stellte ich jedoch fest, daß ich beim Herausnehmen der Karte aus Versehen den Joystick-Stecker berührt und dabei einen Fehlkontakt ausgelöst hatte. Dieser Fehlkontakt erzeugte einen ständigen „offenen Apfel". Nachdem der Stecker wieder fest in seinem Sockel saß, war das Problem behoben.

Z-RAM für den Apple IIc

Ein Erfahrungsbericht

von Ulrich Tönnies

1. Gegenstand

Das Z-RAM-Board der amerikanischen Firma Applied Engineering stellt eine CP/M-Karte mit zusätzlicher Speichererweiterung dar. Es ist ausgestattet mit einem Z80-Prozessor sowie RAM-Chips für eine Speichererweiterung um 256K bzw. 512K (die 256K-Version kann durch Einstecken von geeigneten Chips auf 512K erweitert werden). Zum Lieferumfang gehören ein Handbuch sowie diverse Software. Die folgenden Angaben beziehen sich auf die 512K-Version von Z-RAM.

2. Installation

Der Einbau von Z-RAM in den Apple IIc geschieht anhand der ausführlichen Anweisungen des (englischen) Handbuchs. Bis auf die im folgenden genannten Punkte ist er problemlos zu bewältigen. Zunächst ist das IIc-Gehäuse nach Lösen der 6 Gehäuseschrauben zu öffnen. Nach Entfernen des Rück-Panels des IIc sollte man dann nicht versuchen, die „Nase" am vorderen Ende des IIc aufzuhebeln, wie im Z-RAM Handbuch beschrieben (geht ziemlich schwierig), sondern besser erst zwei „Rasten" am Disketteneinschub (vgl. Abb. 1: Disketteneinschub*) aufhebeln, das Oberteil des IIc-Gehäuses ist dann ganz leicht hochzuklappen. Danach kann man die Tastatur zur Seite ziehen.

Es werden dann die 65C02-CPU und der MMU-Chip des IIc aus ihren Fassungen gezogen und in entsprechende Fassungen auf dem Z-RAM-Board gesteckt. Danach ist das Z-RAM-Board in die freigewordenen Sockel der IIc-Platine zu stekken. Das geht u.U. zunächst scheinbar nicht, es „wackelt". Einen hilfreichen Hinweis hierzu findet man leider erst am Ende des Z-RAM-Handbuchs: Apple „vergißt" bei der Montage des IIc wohl ganz gern, einige Plastikstreifen aus CPU- und MMU-Fassung zu entfernen. Hat man dies dann selbst erledigt, so paßt Z-RAM aber genau.

Schließlich ist noch das eingebaute Laufwerk des IIc zu lösen, an den darunter befindlichen TMG-Chip ist mittels eines „Clips" das Z-RAM anzuschließen. Beim Einbau des Laufwerks achte man darauf, die Laufwerksschrauben nicht zu fest „anzuknallen", das könnte wegen des „auftragenden Clips" zu Platzproblemen mit dem Steppermotor des Laufwerks führen. Schließlich baut man den IIc wieder zusammen und nimmt Z-RAM in Betrieb.

3. Einsatzmöglichkeiten von Z-RAM

3.1. CP/M

Mit dem Board kommt auch das Betriebssystem CP/AM 4.0, mit welchem sich auch das ca. 50seitige Handbuch neben der Installationsanweisung vorwiegend beschäftigt. Ein Manko dieses Betriebssystems ist, daß es im Lieferzustand nicht an den IIc angepaßt ist: Mittels CONFIGIO hat man die Terminalemulation und Tastaturdefinition selbst vorzunehmen.

Neben dem Betriebssystem wird noch RAM-Disk-Software mitgeliefert. Man kann damit ein Laufwerk C: mit einer Kapazität von 576K simulieren. Außerdem für die „CP/M-Welt" gedacht sind noch einige Utilities sowie Patchprogramme für Wordstar V3.3x, mittels deren man Wordstar an den IIc anpassen kann. Erste Tests mit gängigen Programmen zeigten, daß keine allzugroßen Probleme zu erwarten sein dürften.

Der Z80 ist offenbar mit 2MHz getaktet. Geschwindigkeitsvergleiche zu „gängigen" CP/M-Karten zeigten keine wesentlichen Unterschiede (man sollte sich z.B. durch die Aussagen des Pandasoft-Ergänzungskatalogs nicht auf 4MHz einrichten). Die Karte bootet zudem CP/M 2.23, nicht aber CP/M 2.20. Programme, die auf Disketten mit CP/M 2.20 vorliegen, sind aber nach Kopieren des CP/AM 4.0 auf diese Disketten in aller Regel lauffähig.

3.2. DOS 3.3

Für DOS 3.3 kann man ebenfalls die Speichererweiterung als RAM-Disk nutzen. Es werden die Laufwerke S3,D1-D3 simuliert mit 744 bzw. 749 nutzbaren Sektoren. Ein für diese Situation gepatchtes FID wird mitgeliefert. Außerdem läßt sich DOS 3.3 durch den SPEEDOS-Patch (im Lieferumfang) „schnell" machen. Probleme dürften auch hier nicht zu erwarten sein, z.B. lief der gegen DOS-Patches sonst recht anfällige TASC-Compiler mit SPEEDOS auf der RAM-Disk einwandfrei. Die Benutzung von Z-RAM unter DOS 3.3 wird im Handbuch nicht weiter erläutert, jedoch durch ein menügeführtes Programm leicht ermöglicht.

3.3. ProDOS

Hier gelten sinngemäß die Ausführungen zu DOS 3.3. Es ist möglich, nicht den gesamten Zusatzspeicher von Z-RAM für eine RAM-Disk heranzuziehen, sondern nur einen Teil (64K-Banks sind „sperrbar"). Die RAM-Disk-Software unter ProDOS zieht auch nicht die „oberen" 64K des IIc, wo die Standard-ProDOS-RAM-Disk installiert wird, zum Speichern heran. Dadurch können Programme, die die 128K der Mutterplatine des IIc nutzen, weiterhin eingesetzt werden, z.B. die Programme der Utility-Sammlung EXTRA-K von Beagle Bros. Das darin enthaltene Programm Extra.Variables ermöglicht z.B. einen Speicherplatz von ca. 35.000 Bytes für das Applesoftprogramm und zusätzlich ca. 60.000 Bytes für Variablen, und Z-RAM bietet dann die 512K RAM-Disk.

*Bezieht sich auf Abbildung im Z-RAM-Handbuch. Vgl. die 4 Abbildungen in Peeker, 2/86, S. 63 zur Cirtech-Z-80-Karte für den IIc.

Peeker-Sammeldisk # 18

Einzelpreis DM 28,–; Fortsetzungspreis DM 20,–
Achtung:
Diskette hat sowohl UCSD-Pascal- als auch DOS-3.3-Directory.

DISK18:

CALLDUMP.TEXT	40	16-May-86	6
CALLDUMP2.TEXT	26	16-May-86	46
SUPERDUMP.TEXT	14	16-May-86	72
SUPERDUMP.LIB	10	16-May-86	86
EPSON.TEXT	4	16-May-86	96
EPSON.CODE	2	16-May-86	100
IMAGEWRITR.TEXT	4	16-May-86	102
IMAGEWRITR.CODE	2	16-May-86	106
LIB.TEXT	4	16-May-86	108
FUELLER	168	16-May-86	112

A 002 PEEKER === HEFT 05/86 ===
A 002 A–
B 022 EDIT
A 002 B–
A 003 MDB.KOPY.SPEZIAL
A 004 MDB.KOPY
T 016 T.MDB.KOPY.OBJ
B 003 MDB.KOPY.OBJ
A 002 PEEKER === HEFT 06/86 ===
A 002 C–
T 002 PLOT.3.PRO
A 059 PLOT.3.E
B 007 PLOT.BX
T 055 T.PLOT.BX
T 015 PLOT.HELP.1
T 012 PLOT.HELP.2
T 011 PLOT.HELP.3
T 011 PLOT.HELP.4
T 008 PLOT.HELP.5
T 009 PLOT.PATCH
T 005 SUPERDUMP.PATCH
A 002 D–
A 011 PRODOS.BACKUP
T 024 T.PRO.BACKUP.0
B 004 PRO.BACKUP.0
A 005 KOPY.160.SPUR
T 018 T.KOPY.160.SPUR.0
B 003 KOPY.160.SPUR.0
T 001 DISK HAT AUCH PASCAL-DIRECTORY

Hüthig Software Service
Postfach 10 28 69 · 6900 Heidelberg 1

3.4. Appleworks

Durch ein „Expander"-Programm kann man die Fähigkeiten von Appleworks beträchtlich erweitern. So läßt sich Appleworks z.B. komplett in den Z-RAM-Speicher laden. Man kann einen internen Drucker-Puffer installieren und geeignete „Clocks" abfragen (Applied Engineering liefert z.B. eine für den IIc). Wesentlich dürfte auch sein, daß sich die Zahl der Einträge/Zeilen in Textverarbeitung, Rechenblatt und Datenbank vervielfacht. Die „Schreibtischkapazität" von Appleworks kann auf über 400K vergrößert werden, entsprechend große „Dokumente" können sukzessiv auf mehreren Disketten (zu je 143K) gespeichert werden. Wichtig: Die Erweiterung von Appleworks funktioniert nur mit der neuen Version V1.2; wer noch die ältere Version besitzt, sollte versuchen, sie beim Händler „upzudaten".

3.5. UCSD-Pascal

Die im Lieferantenkatalog (Pandasoft) und auch in aktuellen Anzeigen von Applied Engineering in US-Zeitschriften genannte Software für Pascal 1.1 und 1.2 fehlte bei Lieferung. Auf einen Anruf hin erklärte Pandasoft dies durch eine von Applied Engineering nicht angekündigte Änderung des Lieferumfangs. Jedoch schickte Pandasoft die fehlende Pascal-Software innerhalb von 10 Tagen kostenfrei zu.

Die RAM-Disk-Software läßt sich unter Pascal 1.1 und 1.2 nutzen, die folgenden Ausführungen beziehen sich auf die 128K-Version von Pascal 1.2. Simuliert wird ein Volume „RAM:" oder „#12:" mit 508K = 1016 Blöcken. Die Zuordnung zu „#12:" bzw. den Volumenamen kann man leicht ändern, da die dafür verantwortlichen Programme im Sourcetext beigefügt sind.

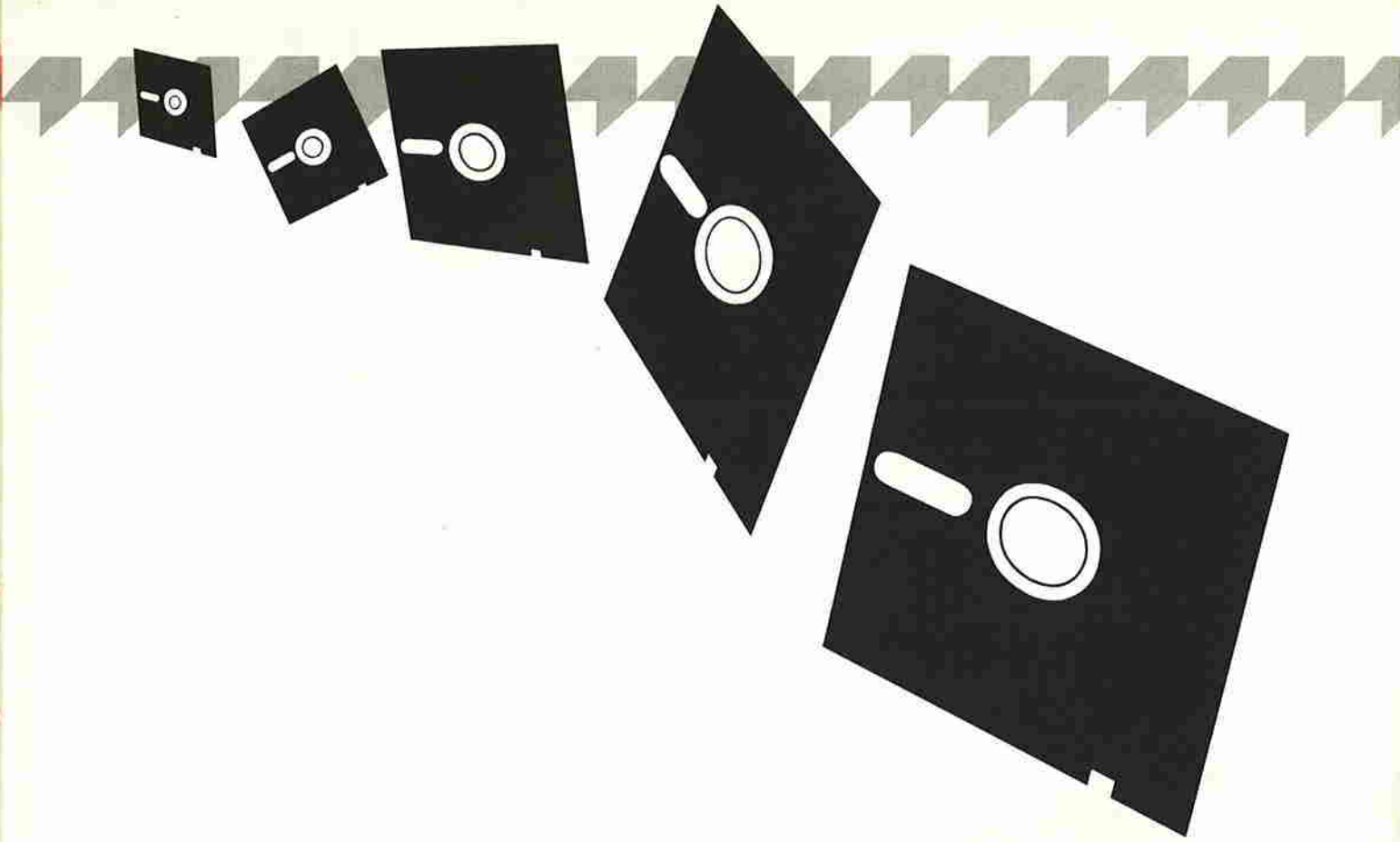
Die Installation der RAM-Disk läuft etwa wie folgt: Beim Booten wird zunächst ein geeigneter Driver installiert, das RAM-Volume initialisiert und das Startup-Programm geladen. Dann werden per Exec-File z.B. sämtliche System-Files auf das RAM-Volume gebracht (man kann den Exec-File leicht nach eigenem Geschmack ändern oder ergänzen). Schließlich kommt der eigentliche „Clou": Das RAM-Volume wird zum Boot-Volume deklariert. Ab jetzt braucht man eigentlich kein weiteres „reales" Laufwerk mehr, besonders der sonst doch häufige Diskettenwechsel ist nun kein Problem mehr. Eine ganz nette Erscheinung ist, daß die Compilierungszeiten deutlich kürzer werden (aber bitte keine Wunder erwarten: Wenn man ohnehin schon unter Pascal 1.2-128K arbeitet und damit auf (*$S+*) weitgehend verzichten kann, so ist nur das während des Compilierens häufige Anlaufen der Laufwerke eliminiert, der Compiler selbst ist nicht schneller als sonst auch).
Probleme gab es beim Betrieb unter Pascal bisher keine, es wirkt nur auf die Dauer etwas ärgerlich, daß z.B. bei Benutzung von System.Charset anscheinend die „Suchreihenfolge" etwa wie „#4:", „#5:", ..., „#12:" aussieht, d.h. daß immer sämtliche vorhandenen „realen" Laufwerke zuerst einmal anlaufen. Das ist jedoch kein Kritikpunkt an Z-RAM, es beruht wohl eher auf der Implementation des Pascalsystems. Dadurch kann es passieren, daß man beim Testen von Programmen zwar auf dem Boot-Volume „#12:" z.B. die umgeschriebene Version einer Utility hat, aber zunächst, ohne es zu bemerken, immer noch eine „alte" Version von einer Diskette gelinkt bekommt. Eigene Utilities kann man ohne weiteres so umschreiben, daß der „Suchvorgang" auf „#12:" beginnt.

3.6. Zusammenfassung

Beim ersten Aufruf der RAM-Disk-Software unter einem bestimmten Betriebssystem wird die RAM-Disk jeweils frisch initialisiert. Wird später nochmals die Installationssoftware aufgerufen, so wird in der Regel ein intaktes RAM-Laufwerk nur wieder „angeschlossen", aber nicht neu initialisiert, d.h. daß z.B. nach Reset etc. die Daten der RAM-Disk nicht unbedingt „weg" sein müssen. Man kann sogar z.B. ein Pascal-RAM-Volume halten, zwischendurch DOS 3.3 booten (natürlich hier dann keine RAM-Disk benutzen) und nachher erneut Pascal booten: Das alte RAM-Volume ist noch intakt. Natürlich kann man auch eine normale Neuinstallierung (mit sog. „Init") auslösen, indem man die „Solid-Apple"-Taste beim Aufruf der Installationssoftware gedrückt hält. Drückt man dagegen zum genannten Zeitpunkt „Open-Apple", so bekommt man durch unterschiedliche „Piepstöne" den Zugriff auf die RAM-Disk angezeigt.

4. Schlußbemerkung

Z-RAM für den Apple IIc ist keine ganz billige Erweiterung, aber m.E. eine recht nützliche. Die Hard- und Software scheint ausgereift zu sein. Ein einzelner Benutzer dürfte jedoch lange brauchen, um alle Einsatzmöglichkeiten der Karte zu nutzen. Gerade darin liegt wohl auch der Wert der Karte, die eben nicht nur eine „CP/M"-Karte ist, sondern ein recht universell einsetzbares Werkzeug für alle wichtigen z.Zt. auf dem Apple laufenden Betriebssysteme. Es sei noch angemerkt, daß es Vergleichbares auch für den Apple IIe gibt: Unter dem Namen RAM-Works wird eine Karte verkauft, die m.W. bis zu 3M ausbaufähig ist.



Analytisches Inhaltsverzeichnis der Peeker-Sammeldisketten #1 bis #18

Wenn Sie wissen wollen, was sich auf *einer bestimmten* Sammeldisk *insgesamt* befindet, so schlagen Sie nach für die Disketten
#1 bis #11 in Heft 11/85, S. 71-78,
#12 bis #15 in Heft 4/85, S. 47,
#16 + #17 in Heft 5/85, S. 74 + 68.
Korrekturen zu den Programmen finden Sie in Heft 11/85, S. 71ff. sowie 5/86, S. 73ff. Das Aufsatz-Jahresinhaltsverzeichnis (Heft 1/84 bis 12/85) ist auf Sammeldisk #12 mit Stichwort-Suchprogramm enthalten.

Ältere Disketten können Sie separat zum Sonderpreis von je DM 20,- nachbestellen,
1. wenn Sie momentan Sammeldisk-Fortsetzungsbezieher und auf Ihrer Bestellung die Kundennummer vermerken, oder
2. wenn Sie kein Fortsetzungsbezieher sind, aber rückwirkend mindestens 6 Sammeldisketten – beliebig gemischt – bestellen.
Ansonsten gilt der normale Einzelpreis von DM 28,- je Sammeldisk.

Das analytische Inhaltsverzeichnis nennt den Zweck des jeweiligen Programms, gefolgt von der Sammeldisk-Nummer und der Heft-Nummer. Beispielsweise bedeutet „#10-10/85/22" = „Programm auf Disk # 10 mit Beitrag in Peeker, Heft 10, 1985, Seite 22".

1. Hardware

1.1. II+/e/c allgemein

Neue ROMs IIe: Diverse Demoprogramme: #10-10/85/22
IIc-Schnittstellen: Initialisierungsprogramm: #14-2/86/34
IIe/c-Maus: Prozeduren für UCSD-Library: #4-4/85/48

1.2. Drucker
(s.a. Bildschirm)

Epson-FX80: Hires-Grafik-Ausdruck: #1-1/84/40 + #5-6/85/22 (DOS 3.3) + # 18-6/86/40 (UCSD)
Imagewriter-I: Hires-Grafik-Ausdruck: #5-6/85/22 (DOS 3.3) + # 18-6/86/40
Imagewriter-I: Steuerbefehlsprogramm: #12-12/85/8
Imagewriter-I: Sonderzeichensatz-Programm: #16-4/86/32
Olympia-ESW100-RO: Hires-Grafik-Ausdruck: #6-6/85/34
NEC 8023: Hires-Grafik-Ausdruck: #10-9/85/30
Großformatiger Grafik-Ausdruck (mit grafikfähigen Interface): #11-11/85/20

1.3. Bildschirm

40-Z/Z-II+/e/c: Speichern des Bildschirms (DOS 3.3): #2-1/85/42
80-Z/Z-IIe/c: Speichern + Ausdrucken des Bildschirms (DOS 3.3): #4-4/85/33 + #4-4/85/76 + #16-4/86/52
80-Z/Z-IIe/c: VTAB-HTAB-Demo: #5-5/85/69
80-Z/Z-IIe/c: Ausdrucken des Bildschirms (UCSD): #12-12/85/48
80-Z/Z-IIe/c: Schnelle Bildschirmroutine für UCSD-Library: #17-5/86/29
II+/e/c: Speichern des Bildschirms in Language-Card (DOS 3.3): #14-2/86/40
Videx-Ultraterm: Zeichensätze: #1-2/84/60
Videx-Karte: Speichern des Bildschirms: #2-1/85/42
Videx-Zeichensatz: EPROM-Programm: #7-7/85/29

1.4. RAM-Karten und RAM-Disk-Driver

16K-Language-Card-RAM-Disk-Driver (DOS 3.3): #2-1/85/8
IIe/c-64K-RAM-Disk-Driver (Pascal 1.1): #6-6/85/48
IIe-64K-RAM-Disk-Driver in Assembler + Turbo-Pascal (CP/M 2.20, Z80): #6-6/85/55 + #12-12/85/62
Balfer-VIA-Karte: Slot-RAM-Disk-Driver (DOS 3.3) für IIe-64K-Karte: #5-5/85/12
AP20-256K-RAM-Disk-Driver (DOS 3.3): #2-1/85/18
AP33-512K-RAM-Disk-Driver (ProDOS): #7-7/85/8

1.5. Sonstige Hardware

Accelerator-IIe: Diverse Utilities: #1-1/84/19 + #4-3/85/66
Wildcard-Plus: Softswitch-Tests: #4-3/85/72
Modem-Karte: Terminal-Programm: #4-4/85/34 (Patch für SSC: 6/85/72)
S.A.M.-Sprachkarte: Vorleseprogramm: #7-7/85/71
EPROM-Programmiergerät: EPROM-MER-Programm: #10-10/85/40

2. DOS und Disk-Drives

(s.a. RAM-Karten)

Betriebssystemunabhängiges Formatierungsprogramm (35-80 Spuren): #7-7/85/20
Hardware-Testprogramm für Disk-II-Drives: #8-8/85/7
Programm zur Ermittlung defekter Sektoren: #11-11/85/24

2.1. Konvertierungsprogramme

GETPAS: UCSD → DOS 3.3: #2-1/85/68
GETDOS: DOS 3.3 → UCSD: #2-1/85/70 + #15 (verbesserte Version)
GETCPM und WS.TRANSFER: CP/M → DOS 3.3: #4-3/85/35
(Für DOS 3.3 → CP/M nehme man APDOS von CP/M-Systemdisk)
PASTOPRO: UCSD → ProDOS: #7-7/85/62
PROTOPAS: ProDOS → UCSD: #9-9/85/58
PROTODOS: ProDOS → DOS 3.3: #14-1/86/36
(Für DOS 3.3 → ProDOS nehme man DOSTOPRO von „Apple-ProDOS, Bd.2")
READPAS: UCSD → CP/M-Turbo: #14-1/86/48

2.2. DOS 3.3

Diversi-DOS 2C als Datei: #6-6/85/74
Diversi-DOS 4C als Datei: #7-6/85/72
Harddisk-Backup-Programm (DOS 3.3; u.a. Megaboard): #18-5/86/61
File-Manager-Grundbefehle: #5-5/85/6
Neues FID-Dateikopierprogramm: #10-9/85/12
DOS-3.3-Mover für Language-Card: #14-2/86/17

2.3. ProDOS
(versionsabhängige Programme gesondert vermerkt)

Diskettenkopierprogramme (1.0.1): #2-1/85/22
Dateikopierprogramme (1.1.1): #8-8/85/18
Harddisk-Backup-Programm (ProDOS, u.a. Megaboard): # 18-6/86/30

Disk-Editor: # 17-5/86/8 + #18-6/86/24
Directory-Einleseprogramme: #4-4/85/41
Directory-Komprimierungsprogramm: #15-3/86/20
Dateileseprogramm (TYPE-Befehl): #4-4/85/46
Schnelles Boot-Programm für ProDOS: #10-9/85/20
Block-Tracer (nur 1.0.1): #5-5/85/51
FORMAT-Befehl (nur 1.0.1): #5-5/85/56
Kompatiblen-Patch-Programm: #2-1/85/31 (1.0.1) + 5/86/59 (1.1.1; nicht auf Sammeldisk!)
Erphi-160-Spur-Patch-Programm für „Superquick": #17-5/86/59
Ehring-160-Spur-Patch-Programme (1.0.1): #11-11/85/29

2.4. UCSD-Apple-Pascal

Directory-Duplizierungsprogramm: #2-1/85/64
Datei-Aufbau-Analyse-Programme: #15-3/86/48 + #17-5/86/34

3. BASIC

(s.a. Grafik)

3.1. Mac-BASIC und MBASIC

Vierteiliger Macintosh-Microsoft-BASIC-Kurs: Demoprogramme (nicht auf Sammeldisk!): 2/84/54 + 1/85/78 + 4/85/63 + 8/85/60
Z80-MBASIC-Kompaktkurs: Befehlsbeispiele (nicht auf Sammeldisk!): 9/85/48

3.2. Applesoft-Befehlserweiterungen

INSTRING-Befehl: #1-2/84/40
DELETE-Array-Befehl: #1-2/84/51
FRE-Befehl (Garbage-Collection): #2-1/85/33
INPUT-Anything-Befehl: #4-4/85/70 + 5/85/50
INPUT-Befehl mit Editierkommandos: #12-12/85/20
RANDOM-Befehl: #5-5/85/69
PRINT-USING-Befehl: #11-11/85/49
QUICKSORT-Befehl: #13-1/86/16
Quicksort-Demos (Applesoft): #13-1/86/6
Befehlssammlung (&GOTO, &GOSUB, &RESTORE, &SOUND, &SCROLL u.a.): #6-6/85/43
CHRGET: Befehlserweiterungsroutine: #14-2/86/38
(Hinweis: Komplettsammlung aller Interpreterroutinen in Heft 12/85, 30-60; nicht auf Sammeldisk!)

3.3. Applesoft-Programmierhilfen

Applesoft-Zeileneditor (DOS 3.3) mit Renumber, Variablen-Referenz u.a. (nur für Fortsetzungsbezieher!): #16-4/86/13
Referenz-Prüfung (GOTOs, GOSUBs): #11-11/85/55
Generator für Bildschirm-Menüs (Applesoft): #4-4/85/20
Konvertierung von Klein- in Großbuchstaben: #10-9/85/31

4. Pascal

4.1. UCSD-Apple-Pascal

Pascal-Grundkurs: Übungsprogramme: #10-12/85/33
Transzendentale Funktionen für Library: #5-5/85/35
IDSEARCH-Befehl: #8-8/85/48
P-Code-Komprimierungsprogramm: #10-10/85/50
UCSD-6502-I/O-Demos: #14-2/86/48
Funktionseingabe-Programm: #16-4/86/58

4.2. Kyan-Pascal

Grundkurs: Übungsprogramme: #14-2/86/56
Aubaukurs: Übungsprogramme + String-Include-Datei: #15-3/86/32
Kyan-Assembler: Übungsprogramme: #16-4/86/48

4.3. Turbo-Pascal

Pascal-Grundkurs: Übungsprogramme: #10-12/85/33
Fußballweltmeisterschaft: Simulationsprogramm: #17-5/86/6

5. Assembler

5.1. 6502/65C02

6502-Kurs: Demoprogramme (nicht auf Sammeldisk!): 7/85/33
65C02-Disassembler: #1-1/84/14
65C02-Makros für 6502-Assembler: #4-4/85/30
Disassembler für undokumentierte 6502-Opcodes: #5-5/85/31
32-Bit-Hex-Dez-Konvertierung: #7-7/85/69
Takt-Zähler-Programm: #10-9/85/32
(Hinweis: Komplettsammlung aller legalen Monitorroutinen II+/e/c in Heft 5/85/52; nicht auf Sammeldisk!)

5.2. Z80

Z80-Kurs: Demoprogramme (nicht auf Sammeldisk!): 11/85/33

5.3. 68000

68000-Disassembler für Mac (in MS-BASIC): 5/85/41; gesonderte 3,5-Zoll-Disk, solange Vorrat reicht.
68000-Kurs: Demoprogramme (nicht auf Sammeldisk!): etwa 7/86

6. Grafik

(s.a. Drucker)

6.1. Applesoft/6502-Programme

Grafik-Demos für Anfänger: #8-8/85/67 + 13-1/86/61
Turtle-Grafik in Basic: #1-1/84/26
Funktionsplotter für einfache Hires: #5-5/85/17
Funkionsplotter für doppelte Hires: #18-6/86/6
Dreidimensionaler Funktionsplotter: #14-2/86/43
Hires-Grafik-Editor mit Zirkelbefehlen u.a.: #10-10/85/6
Sprite-Editor für Animationen: #11-11/85/6
Textausgabe in Hires-Grafik: #12-12/85/16
Konvertierung Double-Hires → Hires: #5-5/85/25

6.2. Applesoft-Befehlserweiterungen

Double-Lores-Grundbefehle: #1-1/84/32
Double-Lores-Befehlssammlung: #8-8/85/32
Double-Hires-Grundbefehle: #1-2/84/24
Interpreter-Änderung für Double-Hires: #10-10/85/14
Pseudo-Double-Hires für II+: #6-6/86/16
XPLOT-Befehl: #4-4/85/14
BOX-FILL-Befehl, SWAP-Befehl u.a.: #6-6/85/6
HSCRN-Befehl, BOX-COPY-Befehl u.a.: #8-8/85/24
Hires-Befehlssammlung (INVERT, TURN, COPY, SWAP, MERGE, SHRINK u.a.): #13-1/86/30

6.3. UCSD-Pascal

Zeichensatz-Generator: #14-1/86/43

7. Mathematik/Technik

Grundrechenarten mit größerer Genauigkeit (Applesoft): #4-3/85/30
Applesoft-Fließkommazahlen: Demo-Programme: #15-3/86/12
Binäre Grundrechenarten: #14-2/86/6 + #16-4/86/6
Primzahlenberechnung (Geschwindigkeitswettbewerbe): #1-2/84/70 sowie #4-3/85/56
Pi-Berechnung bis 15000 Stellen: #17-5/86/42
Fakultätenberechnung (bis max. 9999!): #8-8/85/56
Fourier-Analyse: #6-6/85/38
Meßwert-Verarbeitungsprogramm: #15-3/86/58

8. Schule und Beruf

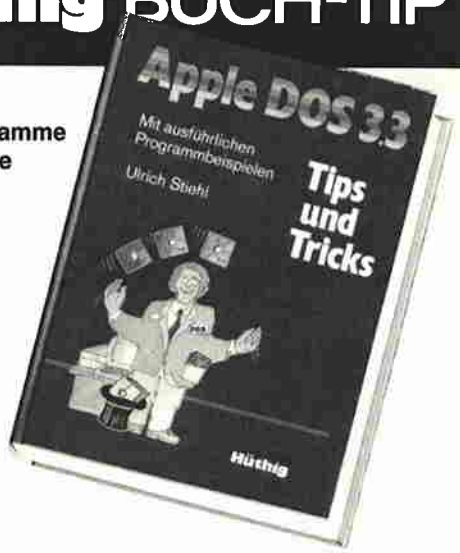
Vokabeltrainer mit Englisch-Grundwortschatz: #13-1/86/20
Legasthenie-Testgenerator: #4-3/85/22
Einkommensteuerprogramm 1984 (MBASIC): #3-1/85/45
Haushaltsverwaltungsprogramm: #16-4/86/42
Simplex-Programm für Betriebswirte: #14-1/86/56
Matrizen-Programm für Betriebswirte: #14-2/86/29
Adreßverwaltung mit dBase II (lauffähig auf dBase V2.4 vom 1.4.83 und CP/M 2.2 sowie 2.26): #8-8/85/40 + 6/86/52

9. Spiele/Musik

Cosmo-Crumble: Weltraum-Abschießspiel: #12-12/85/6
Pyramid-Pitty: Hires-Reaktionsspiel: #7-7/85/6
Solitaire: #6-6/85/64
Puzzle-Spiel: #10-10/85/65
Zeichenjagd (Einzeiler): #8-8/85/69
Tonprogramm (zweistimmig): #16-4/86/60

Hüthig BUCH-TIP

Wegen der neuen Programme paßt die neue Begleitdiskette nur zur 3. Auflage.



Apple DOS 3.3 — Tips und Tricks

von U. Stiehl
3., völlig überarb. Aufl. 1986, X, 203, mit zahlreichen, ausführlich kommentierten Programmlistings, kart., DM 28,–
Begleitdiskette ebenfalls DM 28,–

Dr. Alfred Hüthig Verlag · Postf. 10 28 69 · 6900 Heidelberg 1

Sidekick

Macintosh-Büro-Manager

Borland International (Heimsoeth Software) ist mittlerweile bei Apple-Benutzern durch das Turbo-Pascal zum Begriff für preiswerte Software geworden. Nachdem die Apple-II-Implementierung des Pascal-Interpreters eine weite Verbreitung gefunden hat, bemüht sich Borland nun auch um die Macintosh-Besitzer.

Das für den IBM PC entwickelte Sidekick bietet auf einem Macintosh (Minimalkonfiguration, 128K, eingebautes Laufwerk) eine Reihe von Möglichkeiten, die die tägliche Büroarbeit – und nicht nur diese – erleichtern und neben anderen Anwendungen parallel laufen können (bislang nur als amerikanische Version für DM 260,-). Diese Möglichkeiten verbergen sich unter dem Apple-Pull-Down-Menü und können so jederzeit aufgerufen werden.
Darüber hinaus ist es möglich, einige dieser Punkte auch als reine Anwendungen aufzurufen, die dann noch etwas mehr Komfort und Umfang bieten.
Sidekick enthält u.a.:

- Telefon-Liste
- Termin-Kalender und Wecker
- Arbeitsblätter für Kreditkarten, Geldausgaben und Memos
- Mini-Textverarbeitung
- Taschenrechner
- Terminal-Betrieb

– Die **Telefon-Liste** umfaßt neben Name, Anschrift, Stichwort und Bemerkung auch den Gebührenschlüssel, wodurch während eines Gesprächs, das ebenfalls durch den Macintosh angewählt werden kann (sofern man in den USA wohnt und über das Borland Phone-Link-Interface verfügt), sofort die Gebühren ermittelt und in dem Phone Log, einem persönlichem Telefon-Report, eingetragen werden können. Dadurch ist man jederzeit auf dem laufenden über seine Telefon-Rechnung und – was noch wichtiger ist – wer sie verursacht. Wer nicht in den Genuß kommt, seinen Macintosh als Wählautomaten benutzen zu können, wird sich vielleicht etwas schwertun, bei allen Gesprächen den Rechner zu bemühen, wodurch sich der Stand des Telefon-Reports nicht ganz mit der Rechnung der Bundespost decken dürfte.
Viel wichtiger und praktischer erscheint die Möglichkeit, jederzeit die Daten eines Gesprächspartners auf den Schirm zu bekommen, um somit schnell eine Nummer oder Adresse zu finden oder auch während eines Gesprächs nachzuschauen, mit wem man es gerade zu tun hat. Im Telefon-Report können dann Gesprächsnotizen eingetragen werden. Mit dem sog. Print-Manager kann diese Telefon-Liste mit unterschiedlichem Umfang im Pocket-Format ausgegeben werden.

– Im **Termin-Kalender** können alle Verabredungen eingetragen werden, um sie später auf Knopfdruck parat zu haben. Dabei können sowohl einzelne Tage als auch eine ganze Woche (Week-at-a-Peek) oder ein ganzer Monat überblickt und nach Stichworten abgesucht werden. Auch dieser Kalender kann in Pocket-Format ausgegeben werden.

– Die **Arbeitsblätter** bieten Raum für (jeweils max. 15) Eintragungen
a) über den Kontostand einzelner Kreditkarten,
b) für eine kleine Liste der täglichen Ausgaben und
c) für ein Memo-Blatt, in dem Dinge eingetragen werden können, die man nicht vergessen sollte.
Unter diesem Punkt wird auch der Wecker gestellt, der zum angegebenen Zeitpunkt (Datum, Uhrzeit) ein Alarm-Signal ertönen läßt.

– Die **Mini-Textverarbeitung** bietet die Möglichkeit, kleine Dokumente zu erstellen (z.B. zur Übertragung mit der Terminal) oder vorhandene Textdateien zu editieren.

– Der **Taschenrechner** ersetzt den normalen Taschenrechner im Apple-Menü. Neben den Grundrechenarten werden hier auch die gängigen wissenschaftlichen und einige kaufmännische Funktionen unterstützt. Daneben wird, wie bei einer Rechenmaschine, ein Protokoll der Rechnungen mitgeschrieben, das auch gedruckt werden kann.

– Im **Terminal-Betrieb** können Dokumente über ein Hayes-kompatibles Interface übertragen und empfangen werden, während andere Anwendungen laufen. Auch hier besteht die Möglichkeit des automatischen Anwählens (leider nur in den USA).

Laut Auskunft der Firma Heimsoeth Software ist eine deutsche Version von Sidekick bislang nicht absehbar. An der Möglichkeit des automatischen Wählens wird jedoch gearbeitet. Es bleibt abzuwarten, wie sich die Deutsche Bundespost zu diesem Problem stellt.

Programming Toolkits

für Kyan-Pascal 2.0

Toolkit I: System Utilities: Club-Preis DM 118,-; Normalpreis DM 148,- (lieferbar)
Toolkit II: Mouse Text: Club-Preis DM 118,-, Normalpreis DM 148,- (lieferbar)
Toolkit III: Advanced Graphics: Club-Preis DM 118,-, Normalpreis DM 148,- (Mitte Juni)
Toolkit IV: Turtle Graphics: Club-Preis DM 68,-, Normalpreis DM 88,- (lieferbar)
Toolkit V: Mouse Graphics: Club-Preis DM 158,-, Normalpreis DM 198,- (Mitte Juli)

Alle Utilities werden als teils beidseitig bespielte Disketten geliefert, die neben den Include-Files (meist Quelltexte) diverse Demos enthalten. Die Anleitungen selbst sind Loseblattlieferungen (z.B. bei den System Utilities 52 Druckseiten), die für den grauen Ordner von Kyan 2.0 bestimmt sind. Zum Club-Preis werden nur Mitglieder des Kyan-Clubs beliefert.

Toolkit I: System Utilities

Diese Utilities decken verschiedene Bereiche ab:
1. ProDOS-Utilities: Delete, Rename, Copy, Set-Pefix, Get-Prefix, Lock, Unlock, Make-Directory, Get-Directory, Remove-Directory, Scan-File, Format, Print-File, Bsave, Bload, Set-Time, Get-Time, Set-Date, Get-Date, Get-Clock, Set-Clock, Find-Clock.
2. Maus und Joystick: Find-Mouse, Init-Mouse, End-Mouse, Home-Mouse, Mouse-Click, Mouse-held, Mouse-moved, Mouse-x, Mouse-y, Set-Mouse-xy, Set-x-Bounds, Set-y-Bounds, Print-Mouse-Char, Button-0, Button-1, Joystick-x, Joystick-y.
3. Bildschirmsteuerung (funktionieren teilweise nur auf IIe/c): Clear-Screen, Clear-Line, Clear-End-of-Line, Clear-End-of-Page, Inverse, Normal, Tab, Scroll-up, Scroll-down, Col-80, On-40, On-80, Screen-Bottom, Screen-Top, Screen-Full, Cursor-x, Cursor-y, Get-Char, Machine-Identification.
4. Zufallszahlen: Random im Bereich min..max, Rnd im Bereich 0..1, Seeding.
5. Zahlenkonvertierungsroutinen: Real – String, String – Real, Integer – String, String – Integer.
6. Sortieren (alphabetisch und numerisch) sowie Mischen (bis zu 5 Files).
7. Line Parsing Routine.

Toolkit II: Mouse Text

Diese Utilities umfassen mehrere Dutzend Befehle für Fenstertechnik usw.
1. Cursor-Befehle: Set-Cursor, Obscure-Curser, Hide-Cursor, Show-Cursor.
2. Interrupts: Check-Events, Get-Event, Post-Event, Set-Key-Event, Flush-Event, Peek-Event.
3. Menü-Befehle: Init-Menu, Set-Menu, Menu-Select, Menu-Key, High-Light-Menu, Disable-Menu, Disable-Item, Check-Item, Set-Mark.
4. Kontrollbefehle: Find-Control, Set-Control-Max, Track-Thumb, Update-Thumb, Activate-Control.
5. Fensterbefehle: Init-Window-Margin, Close-Window, Open-Window, Find-Window, Front-Window, Select-Window, Drag-Window, Grow-Window, Screen-to-Window, Window-to-Screen, Close-all, Window-Char, Window-String, Window-Block, Window-Text, Window-Op.

Toolkit IV: Turtle Graphics

Diese Diskette enthält diverse Hires- und Ton-Routinen.
1. Turtle-Befehle: Init-Turtle, Turtle-x, Turtle-y, Turtle-Angle, Graf-Mode, Text-Mode, Pen-Color, Turn, Turn-to, Move, Move-to, View-Port, Full-Port, Fill-Port, Fill-Area, Save-Hires, Load-Hires.
2. Ton-Befehle: Beep, Note, Clock, Phaser.
3. Balkendiagramme usw.: Bar-Chart, Pie-Chart, Plot-x-y.

Hüthig Software Service
Postfach 10 28 69 Heidelberg

Z80+ Card

getestet von Hans-Martin Eng

Wie der Name schon andeutet, handelt es sich bei der Z80+ Card um eine Z80-Coprozessorkarte für den Apple II+ und IIe. „Noch eine Z80-Karte" sagen da vielleicht einige Leser, doch gibt es da einige Punkte, die diese Appleerweiterung gegenüber anderen CP/M-Karten auszeichnen. Ich will jedoch dem Testergebnis nicht vorgreifen.
Zum Lieferumfang der Z80+ Card gehören zum einen die Karte selbst, dann eine Diskette mit diverser Software (allerdings ohne CP/M-Betriebssystem!) und zuletzt ein Handbuch im DIN A4-Format. Beginnen wir mit der Hardware:

1. Die Z80+ Card selbst

Auf der 22cm * 7cm großen Platine befinden sich, neben einer Z80H-CPU, 64K RAM und diverse Steuerbausteine. Der Systemtakt wird von einem 16,000 MHz-Quarz erzeugt. Am linken Ende der Karte fallen noch ein Jumperblock und drei LED auf. Bis auf drei sind alle ICs in gedrehten Sockeln untergebracht, was eine hohe Kontaktsicherheit gewährleistet.
Alle Bauelemente sind sauber verlötet, aufgrund der hohen Pakkungsdichte der Karte mußten die Leiterbahnen jedoch reichlich dünn ausfallen. Es ist also Vorsicht geboten, wenn man mit spitzen Gegenständen operiert.
Die Z80+ Card ist die einzige mir bekannte CP/M-Karte, die sowohl zur weit verbreiteten Microsoft-Softcard als auch zur ALS-CP/M-Card kompatibel ist. Sie kann somit sowohl CP/M 2.20/2.23 als auch ALS-CP/M 3.0-Software einwandfrei verarbeiten. Dies geschieht in zwei getrennten Betriebsmoden.
Im Betriebsmodus 0, angezeigt durch eine leuchtende gelbe LED, arbeitet die Z80+ Card wie die normale Softcard – leider auch nur mit 2 MHz Takt. Sie greift dabei auf den normalen Arbeitsspeicher des Apple zu, der Speicher der Karte selbst kann als RAM-Disk verwaltet werden. In diesem Betriebsmodus laufen alle CP/M-Programme für den Apple ohne Einschränkungen. Ob der Speicher der Z80+ Card auch für den 6502 des Apple zugänglich ist, darüber schweigt sich das Handbuch leider aus. (Lt. Rücksprache mit dem Entwickler Dipl-Math. U.Zimmermann geht dies nicht. us)
Im Betriebsmodus 1 legt der Apple jedoch „die Ohren an". Hier wird die Z80H mit 8 MHz getaktet (2 MHz mehr als die ALS-Karte). Sie muß dabei keine zusätzlichen Wait-Zyklen ausführen, so daß diese 8 MHz voll zur Geltung kommen. In diesem Betriebsmodus greift die Z80+ Card nun auf den eigenen Speicher zu. Dies bedeutet, daß alle Programme, die speziell auf den Apple-Speicher zugreifen müssen, etwa für Hires-Grafik o.ä., nicht funktionieren. (Genauer gesagt, sie laufen zwar, erzeugen aber auch keine Grafik.) Hier wären vor allem MBASIC und GBASIC von der Softcard-Master-Diskette zu nennen.
Die Z80+ Card ist mit einem Interrupt-Timer ausgerüstet, der mit ca. 62 kHz Interrupt-Signale für die Z80H erzeugt. Diese Interrupts können über den Jumperblock abgeschaltet werden. Eine Nutzung bleibt wohl auf eigene Anwendungen beschränkt, da für die Nutzung der Interrupts keine Software mitgeliefert wird. Denkbar wären beispielsweise ein Druckerpuffer o.ä..
Die Umschaltung zwischen den beiden Betriebsmoden geschieht über einen Schreibzugriff auf den Softswitch $C080+n0 (n steht für Slot n). Da sich die Karte beim Einschalten im Betriebsmodus 1 befindet, muß man den Softswitch betätigen, bevor man Softcard-CP/M bootet. Ein Patchprogramm, das den Bootprozeß dahingehend modifiziert, daß der richtige Betriebsmodus eingeschaltet wird, befindet sich auf der mitgelieferten Diskette.

2. Die Software

Software für die Z80+ Card wird auf einer 5 1/4"-Diskette mitgeliefert. Auf ihr befinden sich sowohl DOS 3.3- als auch CP/M-Programme. (Auf einer Seite wohlgemerkt.)

2.1. DOS 3.3-Software

Hier handelt es sich um Patch-Software für die CP/M-Versionen 2.20 und 2.23. Das Programm BOOT-PATCH modifiziert, wie schon angedeutet, das BOOT-Programm von CP/M-Disketten; das Betriebssystem selbst bleibt unverändert. So können alle Programme, die unter Softcard-CP/M laufen *uneingeschränkt* weiterverwendet werden.
Nun hat man schon eine 8 MHz-CP/M-Karte, da möchte man den Rechner gefälligst auch mit 8 MHz laufen lassen. Abhilfe bringt hier das Programm CP/M-Patch. Dieses patcht Softcard-CP/M-Disketten so, daß nun alle Software im Betriebsmodus 1 abgearbeitet wird. Dieses gepatchte CP/M nennt sich Z80+-CP/M. (lies: „zet achtzig plus ce pe em".)

2.2. CP/M-Software

Hier bleiben eigentlich keine Wünsche offen. Für den Softcard-Modus wird ein RAM-Disk-Treiber für die 64K RAM der Z80+ Card mitgeliefert. Für das Z80+-CP/M gibt es gleich eine ganze Anzahl verschiedener RAM-Disk-Treiber, dazu noch eine Disk-Utility, mit der man Disketten formatieren und kopieren kann. Man kann damit auch das Betriebssystem auf 40-Spurlaufwerke umstellen. Schließlich kann man das Z80+-CP/M auch an verschiedene gebräuchliche 80-Spur-Controller, wie den Erphi- oder den Ehring-Controller anpassen.

2.3. Das Z80+-CP/M

Dieses CP/M hat es in sich! Der CCP (Console Command Processor) wurde um einige Befehle erweitert:

ALL – ist ein erweiterter DIR-Befehl. Angezeigt werden zusätzlich zur File-Länge auch der Status. Selbst SYS-File werden aufgelistet.
BOOT – ersetzt den COMmand-File gleichen Namens von der Softcard-Master-Disk.
GO – ersetzt ebenfalls einen COM-File der Softcard-Master-Disk. Damit kann man einen COM-File wieder starten, den man vorher verlassen hat. Dieser Trick funktioniert nur, wenn das neu zu startende Programm nicht sich selbst reloziert, wie das z.B. bei allen Debuggern der Fall ist.

Je nach Hauptspeichergröße verfügt man nach dem Kaltstart über 32, 48 oder 110K Speicherkapazität auf einem Pseudolaufwerk P:. Wem das noch nicht reicht, der kann sich eine RAM-Karte zulegen. Für SATURN-, IBS-AP17-, SOLARIM- und CSC-RAM-Karten sind Treiber auf der Utility-Diskette mitgeliefert, wer eine andere Karte besitzt, wird nach Herstellerangaben auch nicht alleingelassen.
Die normale Disketten-Zugriffsgeschwindigkeit setzt dem Faß allerdings die Krone auf. Fragen Sie mich nicht, wie, aber sie wurde um bis zu einen Faktor 4 (!) erhöht.
Da alles so schön klingt, muß ich dem Enthusiasmus einen (allerdings kleinen) Dämpfer verpassen. Es gibt Programme, die unter dem gepatchten CP/M nicht laufen. Der Hersteller nennt hier den Macro-Assembler M80 und das zugehörige CREF80, beide auf der ALDS-Diskette von Microsoft zu finden. Für diese Programme wird ein Patch genannt, der sie wieder zum Laufen bringen soll. Das konnte ich leider in Ermangelung dieser Programme nicht nachprüfen. Ebenso stieg ein mir kurzfristig zur Verfügung gestellter F80-Fortran-Compiler aus. Hier gibt es jedoch eine einfache Abhilfe: Im Softcard-Modus compilieren und linken, die daraus erzeugten COM-Files dürften dann wiederum mit 8 MHz lauffähig sein.
Einwandfrei funktionieren jedoch Wordstar und Turbo-Pascal, auch Multiplan entwickelt eine ungekannte Aktivität. Mit Turbo-Pascal konnte ich auch den vierfachen Geschwindigkeitszuwachs beim Rechnen nachprüfen.

3. Das Handbuch

Dieses besteht aus einem DIN A4-Heft mit Ringheftung (das im Mai neu aufgelegt worden ist, us). Nach einer kurzen Einleitung folgen sehr ausführliche Erläuterungen zu Funktion und Installation der Karte. Weitere 34 Seiten gelten einer detaillierten Beschreibung der mitgelieferten Software. Dann folgen noch Anweisungen, um Fehlerursachen einzukreisen. Der Anhang besteht aus verschiedenen Tabellen.
Das Handbuch ist mit Februar 1986 datiert. Inzwischen war der Hersteller nicht untätig, eine fünfseitige Ergänzung des Handbuchs dokumentiert das. Diese Ergänzung behandelt neue RAM-Disk-Treiber auf der Utility-Diskette für die IBS-AP33 (256K bis 1M) und die OHO-512-RAM-Karte. Außerdem wird ein Patchprogramm für den CAT-Controller mitgeliefert, Turbo-Pascal-Besitzer können sich über einen Include-File freuen, der ihnen wieder die Apple-Grafik zugänglich macht.

Fazit

Die Z80+ Card ist meines Erachtens rundherum gelungen. Das fehlende Betriebssystem wird durch die mitgelieferte Software mehr als wettgemacht. Da man außerdem praktisch über zwei CP/M-Karten verfügt, erscheint mir der Preis von ca. DM 700.-- nicht zu hoch. Vergleicht man diesen nämlich mit dem verschiedener anderer schneller CP/M-Karten, schneidet die Z80+ Card nicht un-

günstig ab. Daß die Grafikbefehle des GBASIC nicht funktionieren, kann man der Karte nicht anlasten. So bleibt eigentlich nur noch der gute Eindruck übrig, den die Karte bei mir hinterließ. Die reichlich mitgelieferte Software macht sie vor allem für Umsteiger interessant. CP/M-Anfänger können sich vielleicht auch bald freuen. Der Hersteller bemüht sich um eine günstige CP/M-Lizenz, die er dann für einen geringen Aufpreis an die Kunden weitergeben will.
Bleibt mir nur noch übrig zuzugeben, daß ich die Karte nach Beendigung des Tests nur äußerst ungern aus meinem Rechner wieder herauszog. (Herr Eng hat sich inzwischen eine Karte gekauft. us)

BASF-Reinigungs- und Justagedisketten

Auf der diesjährigen „Mikrocomputer '86" in Frankfurt stellte die BASF als Neuheit Reinigungs- und Justagedisketten für die neuen 3.5"-Laufwerke vor.
Nach den Angaben des Herstellers erspart vorsorgliches, allwöchentliches Reinigen der Magnetköpfe mit einer Reinigungsdiskette dem Anwender bei geringem Geld- und Zeitaufwand Störungen und Datenverluste.
Die Schreib-/Leseköpfe der Floppy-Disk-Laufwerke gleiten im Betrieb auf der Oberfläche der Diskette entlang. Da dieser Kontakt nie absolut staubfrei erfolgt, können Staub- und Schmutzteilchen auf den Kopf gelangen und Schreib-/Lesefehler verursachen. BASF-Reinigungsdisketten befreien den Magnetkopf von Ablagerungen und nehmen den Schmutz auf. Sie werden als ein- und zweiseitige Disketten angeboten.
Während die Anwendung der Reinigungsdisketten für jeden Floppy-Besitzer problemlos ist, sind die Justagedisketten für Servicetechniker und Fachleute entwickelt worden.
Die BASF-Justage-FlexyDisk wird zusammen mit einem Oszillographen zum Prüfen und Reparieren von Diskettenlaufwerken verwendet. Auf der Justagediskette sind dazu Informationen in Form eines optischen Informationsmusters hochpräzise aufgezeichnet. Am einfachsten lassen sich diese Informationen mit dem Testbild beim Fernsehen vergleichen. Mit deren Hilfe kann der Fachmann durch Messen und Auswerten von Lesesignalen am Oszillographen die Spurlage, den Kopfeinstellwinkel und die Justage des Indexsektors überprüfen und korrigieren. Die BASF-FlexyDisk hat den Vorteil, daß Auswertung, Erkennen der Ursachen und gleichzeitige Korrektur in einem dynamischen Prozeß erfolgen können. Die aufgezeichneten Informationen basieren auf den für die Industrie weltweit gültigen Standards ECMA und ANSI.

Jörg Bliesener beantwortet CP/M- und Wordstar-Leserbriefe

Wenn Sie eine Frage haben, so schreiben Sie uns. Das Problem muß jedoch eindeutig und nachvollziehbar geschildert werden. Also Gerätekonfiguration, Programmversion usw. genau angeben!
Spezialfragen, die wir selbst beantworten können, erledigen wir schriftlich. Fragen, die wir nicht beantworten können und die zugleich von allgemeinerem Interesse sind, veröffentlichen wir im Peeker.

Wordstar-Patch-Tabelle

Benötige vollständige Patch-Tabelle für WS.COM Version 3.0, wo ist diese zu haben?
Armin Klauß, Karlsruhe

Antwort:

Der Wordstar-Installation-Guide sowie die Wordstar-Customization-Notes erhalten Sie bei jedem Distributor von Micropro-Software.

Wordstar und dBase

Ich besitze einen Apple und benutze zum Textverarbeiten „Wordstar" und als Dateiprogramm „dBase II". Im großen und ganzen bin ich mit diesen Systemen zufrieden, doch einige Änderungen würden das Arbeiten noch mehr erleichtern. Zwar besitze ich das Installationsprogramm für „Wordstar", doch gewisse Grundeinstellungen werden dort mit dem Hinweis auf Customization Notes nicht behandelt. Vielleicht können Sie mir behilflich sein.

Wordstar:
1. Sofortiger Aufruf von Drive B: als aktuelles Laufwerk
2. Festlegung linker Rand z.B. 05 und rechter Rand z.B. 75
3. Löschen von „insert on"
4. Änderung der Zeilenzahl pro Wordstarseite
5. Druckausgabe: Bei Aufruf von P muß erst umständlich entweder mit Return oder entsprechend mit Y oder N geantwortet werden. Kann man diese Prozeduren einfach umgehen, oder wie lassen sich diese Voreinstellungen ändern?
6. Wenn ich Wordstar starte, ergeben sich noch folgende Probleme: Erst durch Drücken von 2 x Escape funktioniert Groß- und Kleinschreibung, allerdings mit einigen Einschränkungen. Bei Eingabe von < erscheint Ä, bei > dann ü, bei ? dann ö, bei ü dann ß, bei (dann ä, bei) dann ü und bei ! dann ö. Um die Klammern auf den Bildschirm zu bringen, schalte ich dann einmal Escape und um wieder normal schreiben zu können, drücke ich wieder 2 x Escape.

dBase:
1. Von dBase aus lassen sich mit „set call on" Assemblerroutinen starten. Wie funktioniert das? Welcher Speicherbereich muß benutzt werden, ohne daß das dBase-Programm überschrieben wird etc.?
2. Wie funktioniert die unmittelbare Steuerung von Peripheriegeräten über den Befehl CHR(numerischer Ausdruck)? Ich möchte das Schriftbild meines Druckers über diesen Befehl ändern. Ich habe einen Epson RX-80 Drucker mit Centronic-Schnittstelle.
Ich bedanke mich schon jetzt für Ihre Mühe und hoffe, daß Sie mir in einigen Punkten helfen können.
Horst Peter, Mainz

Antwort:

Zu Wordstar:
1. Sofortiger Aufruf von *Drive B: als aktuelles Laufwerk*
Dies ist mittels eines kurzen Assembler-Programms möglich, das im Wordstar ab Speicheradresse 02E0H installiert wird. Folgende Bytes sind dazu notwendig:
02E0H 0EH
02E1H 0EH
02E2H 1EH
02E3H 01H
02E4H C3H
02E5H 05H
02E6H 00H
02A4H CBH
02A5H E0H
02A6H 02H
Dieses Assembler-Programm wird bei jedem Start des Wordstars aufgerufen und schaltet auf Laufwerk B um.

2,3. Die Adresse der Bytes, in denen der *linke oder rechte Rand* bzw. Insert gespeichert wird, ist:
Label/Adresse/Standardwert
linker Rand INITLM:/037FH/00H
rechter Rand INITRM:/0380H/40H (=64D)
Insert ITITOG:/0362H/FFH
Beachten Sie, daß die Werte des linken und rechten Randes jeweils um eins dekrementiert abgespeichert werden. Der rechte Rand wird automatisch auf den Wert (Bildschirmbreite-4) gesetzt, falls er größer sein sollte. Beim Insert-Flag wird 00H für „Insert aus" und FFH für „Insert an" eingesetzt.

4. Die *Seitenlänge in Zeilen* wird bei Adresse 0367H im Label INITPF:+1 abgespeichert. Jedoch muß für den korrekten Betrieb mit einem Typenraddrucker im Label INITPF:+2 und INITPF:+3 auf Adresse 0368H und 0369H die Anzahl der Zeilen pro Seite multipliziert mit 8 als Wort (Low Byte zuerst) stehen.

5. Die *Abfrage der Werte bei der Druckausgabe* können Sie mit ESC abbrechen. Dabei werden dann für die noch nicht abgefragten Werte die Standardeinstellungen eingesetzt. Wenn Sie diese ändern wollen, so ist das ab Adresse 03CAH (Label PODBLK:) möglich:
Label/Adresse/Standardwert
Diskettenausgabe
PODBLK:/03CAH/00 (= nein)
Form-Feeds benutzen
PODBLK:+1/03CBH/00 (= nein)
Seitenform unterdrücken
PODBLK:+2/03CCH/00 (= nein)
Pause zwischen Seiten
PODBLK:+3/03CDH/00 (= nein)
Auch hier gilt wieder:
DFFH = ja, 00H = nein

6. Das *Tastatur-Problem.* Die Programmierer von Micropro haben sich durchaus etwas einfallen lassen, um dem Apple die Groß- und Kleinschreibung beizubringen. Dazu gehört das von Ihnen geschilderte Problem. Wahrscheinlich erscheint beim Starten Ihres Wordstars auch die Meldung „Videx with Software U/L-Case" oder so ähnlich. Das bedeutet, daß der Wordstar softwaremäßig die Groß/Kleinschreibung erzeugt, die die Original-Apple-Tastatur ja nicht kann. Einmal ESC drücken bedeutet dann, daß das nächste Zeichen in Großschreibung umgewandelt wird, zweimal schaltet in den „Shift-Lock"-Modus um. Weiterhin kann der Original-Apple-II keine Umlaute. Diese werden ebenfalls im „Shift-Lock"-Modus erzeugt, und zwar genau mit den von Ihnen erwähnten Tasten. Das nützt zwar Besitzern eines Apple II oder II Plus mit Original-Tastatur, IIe-Benutzer oder Leute, die eine externe Tastatur Ihr eigen nennen, sind aber nicht sehr glücklich darüber. Um auch Ihnen das Gefühl eines „richtigen" Wordstars zu bescheren, sind die folgenden Patches nötig:
Label/Adresse/Wert
APLFLG:/02D7H/05H
/411BH/00H
/41C0H/00H

Zu dBase:
1. Der *Speicherbereich für Maschinenprogramme* liegt ab A000H.
2. Die Ausgabe eines Textes mit der *CHR-Funktion* funktioniert ganz normal, nur ist zu beachten, daß die verschiedenen CHR-Funktionen in einem ?-Befehl nicht durch Kommata getrennt werden, sondern mit dem „+"-Operator zusammengefügt werden, weil dBase sonst nach jedem Datenfeld ein Leerzeichen einfügt. Die Umschaltung auf Kursivdruck funktioniert also so:
? CHR(27)+CHR(52)

Wordstar und Imagewriter
Sie haben in den Peeker-Ausgaben mit „CP/M für Einsteiger" und „MBASIC für den Applesoft-Profi"ein für mich sehr interessantes Thema aufgegriffen. Ich möchte darüber hinaus mehr über das Programmieren des 8080/Z80-Prozessors sowie den Umgang mit dem CP/M-Betriebssystem und Apple-Ports erfahren. Sicher würde dieses Thema auch einen großen Kreis Ihrer Leserschaft interessieren. Mit dem folgenden Problem komme ich leider mangels (für mich) geeigneter Literatur und trotz Erfahrungen mit dem 6502-Prozessor nicht zurecht: Ich möchte Text, den ich mit Hilfe eines Textverarbeitungsprogramms (Wordstar) auf einem Apple II europlus unter dem Betriebssystem CP/M erstellt habe
– auf den Apple-Imagewriter mit serieller Schnittstelle über Slot #1 (wie unter CP/M vorgesehen) ausgeben
– auf einer Schreibmaschine mit paralleler Schnittstelle mit Parallel-Interface über Slot #2 (ist unter CP/M leider nicht vorgesehen) ausgeben.
Wichtig ist für mich dabei natürlich, daß ich das Interface in Slot #1 oder sonstwo nicht wechseln muß (Parallel <–> Seriell). Lieber würde ich unterschiedliche Disketten (d.h. modifizierte Betriebssysteme) für die unterschiedliche Textausgabe verwenden. Ich würde mich freuen, wenn Sie auf diese Fragen in Ihrer Zeitschrift eingehen könnten oder, falls sich diese Angelegenheit mit einem unbedeutenden Patch erledigen lassen sollte, um einen Hinweis im Peeker.
Dipl.-Ing. Winfried Neubert, Berlin

Antwort:
Um den von Ihnen erwünschten Effekt, eine *Datei vom Wordstar aus in Slot 2* ausdrucken zu lassen, zu erreichen, ist es am empfehlenswertesten, den Wordstar zu ändern.
Um eine Druckausgabe in Slot 2 zu bewirken, braucht nur ein einziges Byte geändert zu werden, und zwar an Adresse 071FH (INSTALL-Label LISEND:+2). Normalerweise hat es den Wert 05. Wenn Sie dieses Byte auf 04 ändern, ist alles erledigt.
Zum Ändern brauchen Sie zwei Disketten: Eine mit dem CP/M-Betriebssystem und eine mit Ihrem Wordstar. Das Ändern geht wie folgt vor sich:
CP/M-System in Laufwerk A, auf Laufwerk A umschalten, „DDT" und <Return> tippen. Nach dem Prompt des DDT die Wordstar-Diskette einlegen und „IWS.COM" und <Return> eingeben. Wenn die Datei geladen wurde, geben Sie „S71F" ein und drücken wieder die <Return>-Taste. Nun müßte erscheinen:
071F 05
und der Cursor sollte hinter der 05 stehen. Tippen Sie einfach „04" und <Return>, gefolgt von „." und <Return>. Nun sehen Sie wieder den DDT-Prompt „-". Drücken Sie Ctrl-C, und Sie landen wieder im Betriebssystem. Mit „SAVE 66 WS2.COM" speichern Sie die geänderte Version auf Diskette ab.
Sie haben nun zwei Wordstar-Versionen auf Ihrer Diskette. Die eine, WS, druckt auf Slot 1, die andere, die mit WS2 gestartet wird, gibt die zu druckenden Daten auf Slot 2 aus. Weiterhin müssen Sie die Adresse 0717 (GSWTGH:) auf 01H ändern

---

Wordstar und zwei Drucker
Ich bin im Besitz eines Apple-II-Kompatiblen mit einem Druckerinterface, einer 80-Zeichenkarte und einer Original-Super-Serial-Card. Das Druckerinterface steckt im Slot 1, die Super-Serial-Card in Slot 4. Die 80-Zeichen-Karte befindet sich im Slot 3. Da die Z80-Karte fest eingebaut ist, entfällt Slot 2.
An der Super-Serial-Card ist ein Typenraddrucker angeschlossen. Früher benutzte ich als Textverarbeitungsprogramm den Applewriter, mit dem ich problemlos zwischen Slot 1 und Slot 4 umschalten konnte, um den Matrix- oder Typenraddrucker anzuwählen.
Nun legte ich mir das CP/M-2.2-Betriebssystem (56K-Version 2.20B) und den Wordstar zu. Nach der Anpassung des Wordstar läuft alles problemlos. Nur muß ich immer die Karten im Slot 1 austauschen, um auf dem Matrix- oder Typenraddrucker zu drucken. Um dies zu vermeiden, möchte ich die Super-Serial-Card im Slot 4 lassen.
Da ich sowieso zwei Wordstar-Versionen besitze (eine für den Matrix-, eine für den Typenraddrucker angepaßt), wäre es doch sicher möglich, das BIOS von CP/M so zu verändern, daß der LST-Ausgabevektor nicht Slot 1, sondern Slot 4 anspricht. Dies habe ich bereits versucht, indem ich unter STAT.COM die Adresse DD2C -01 auf DD2C -04 abgeändert habe. Nach dieser Änderung reagierte jedoch die SSC-Karte nicht.
Nun meine Frage an Sie: Können Sie mir vielleicht die Adressen nennen, die ich ändern muß, damit der LST-Befehl nicht Slot 1, sondern Slot 4 anspricht?
Für Ihre Bemühungen recht herzlichen Dank.
Peter Westphal, Olching

Antwort:
Die Änderung des CP/M ist schwierig. Ich würde Ihnen viel eher einen Patch des Wordstars empfehlen. Dieser stellt nämlich die erforderlichen Programme zur Ausgabe von Daten direkt über Ports zur Verfügung. Im Wordstar-Installationshandbuch finden Sie ab Adresse 0728H (INSTALL-Label POBSY:) die entsprechenden Speicherstellen. Die Portadressen der SSC lauten:
Datenport: 0E0C8H (für Slot 4)
Statusport: 0E0C9H
Receive-Ready-Maske: 08H
Transmit-Ready-Maske: 10H

Zur Wordstar-Anpassung im 11/85, S. 61
Trotz verschiedener Versuche, den 2. Zeichensatz (deutsch) auf meiner videxkompatiblen 80-Zeichenkarte für Wordstar zugänglich zu machen (TRMINI, IVON, INISUB), ist dies nicht gelungen (Apple II+, Wordstar 3.0). Welche Werte müssen in den entsprechenden Adressen stehen?
Manfred Schleich, München

Antwort:
Ab TRMINI: sollten die folgenden drei Bytes stehen:
Adresse/Byte
0292H/02H
0293H/1BH
0294H/28H

MBASIC und Slot-Wechsel
Im Peeker, 9/85 haben Sie einen interessanten Artikel zu MBASIC auf dem APPLE gebracht, Autor war Jörg Lange. Einige wertvolle Anregungen konnte ich dem Artikel entnehmen, eine Frage ließ er aber unbeantwortet:
Im Slot 2 meines apple-kompatiblen Rechners (Apple II+) steckt eine serielle Karte, über die ich

Daten aus einem Digitizer einlese. In „Applesoft" geht das einfach über den Befehl „IN#2". Danach erscheinen übertragene Daten auf dem Bildschirm bzw. sind über z.B. „INPUT A$" einlesbar in eine Stringvariable.

In MBASIC gibt es leider meines Wissens keinen Befehl, der die verschiedenen Slots anspricht. „INP" und „OUT" funktionieren ebenfalls nicht, kurzum, ich habe es bisher noch nicht geschafft, in MBASIC Daten über die serielle Karte einzulesen.

Können Sie mir weiterhelfen? Gibt es doch einen entsprechenden Befehl oder ist mit bestimmten Peeks etwas zu erreichen oder...?
Gerhard Vester, Freiburg

Antwort:
Es geht tatsächlich mit Peeks, der Peeker sagt Ihnen, wie: Um ein Byte von der seriellen Karte einzulesen, benötigen Sie folgende Befehlssequenz:
WAIT STATUSPORT, RECVBIT:
A$ = CHR$ (PEEK(DATENPORT))
Zum Ausgeben benutzen Sie den Befehl:
WAIT STATUSPORT, XMITBIT:
POKE DATENPORT, ASC(A$)
Um so etwas wie die INKEY$-Funktion zu simulieren, verwenden Sie:
IF PEEK (STATUSPORT)
AND RECVBIT THEN
A$ = CHR$ (PEEK(DATENPORT))
ELSE A$=""

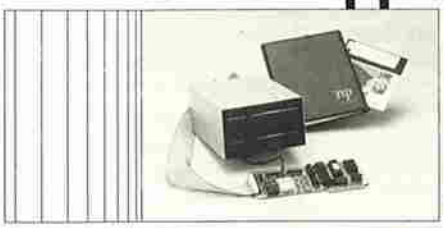
Dabei steht jeweils in A$ das ein- oder auszugebende Zeichen. Natürlich müssen noch die Variablen STATUSPORT, DATENPORT, RECVBIT und XMITBIT initialisiert werden. STATUSPORT gibt dabei die I/O-Adresse des Statusports der ACIA auf der seriellen Karte an, DATENPORT entsprechend den Datenport der ACIA. In RECVBIT ist das Bit gesetzt, das im Statusport den Empfang eines Zeichens anzeigt, entsprechend gilt das gleiche für die Variable XMITBIT, die angibt, wann ein Zeichen gesendet werden kann. Für eine Super Serial Card in Slot 2 sind die Werte folgendermaßen:

STATUSPORT = &HE0A9:
DATAPORT = &HE0A8:
RECVBIT = 8:
XMITBIT = &H10

Wordstar und Drive-Wechsel
Mit Interesse habe ich in Peeker, 3/86 den Artikel über Wordstar-Modifikationen gelesen. Obwohl ich nur Anwender bin, konnte mir der Artikel sehr helfen, denn es ärgerte mich schon länger, daß ich im Eröffnungsmenü viele Änderungen vornehmen mußte (Hilfsgrad auf 0, Blattlänge, Kopfabstand etc.) Jedoch vermisse ich in der Patch-Tabelle die Adresse für „Neues Laufwerk", da ich ständig mit zwei LW arbeite. Könnten Sie mir diese wichtige Adresse mitteilen?
Ferner habe ich folgendes Problem: Ich benutze einen Apple IIe, 2 x 640K-Laufwerke mit Erphi-Controller und einen FX-80 mit APL-B-Drucker-Graphikinterface. Beim Booten von CP/M wird leider der Drucker nicht aktiviert. In der Anleitung zur APL-B wird angegeben, daß das Betriebssystem mit DDT zu ändern ist.
Ich bitte um Mitteilung, ob über die Änderung des CP/M ebenfalls schon ein Artikel in der o.g. Form erschienen ist. Das wäre sehr hilfreich, denn es ist sehr aufwendig, ein bereits gestartetes Programm zum Zweck des DDT-Aufrufs zu verlassen.
Elmar Elsner, Köln

Antwort:
Bezüglich der Einstellung des Default-Laufwerks in Wordstar beachten Sie bitte die Antwort auf den Leserbrief von Herrn Peters (s.o: „Wordstar und dBase").
Zur Modifikation des CP/M, damit ein Drucker erkannt wird, gehen Sie bitte folgendermaßen vor:
Legen Sie die CP/M-Systemdiskette in das Default-Laufwerk und drücken Sie Ctrl-C. Nun starten Sie DDT und geben „S100" ein. Auf die Meldung „100 00 " geben Sie ein: „3E", „31", „32", „2F", „DD", „C3", „00", „00", „.", jeweils gefolgt von <Return>. Nun müßten Sie wieder das Minuszeichen von DDTZ sehen. Tippen Sie Ctrl-C. Nun landen Sie wieder im CP/M. Geben Sie nun ein: „SAVE 1 EPSON.COM". Damit speichern Sie das eben Eingegebene auf Diskette unter dem Namen EPSON.COM ab. Dieses Programm sollten Sie nun nach jedem Booten starten. Das können Sie per Hand machen, indem Sie einfach „EPSON" eintippen, sie können aber auch, wenn Sie das Programm TURNKEY oder AUTORUN haben, EPSON automatisch starten lassen.

Anschlußfertig für Apple® II, //e...



erphi
Doppellaufwerk DL 280
2 Laufwerke
mit je 640 kB formatiert
im Gehäuse incl.
erphi
Autopatch-Controller AFDC 3
Handbuch und Diskette mit Dienstprogrammen
Verkaufspreis incl. MwSt.
DM 1.298,–

erphi
Floppy-Subsystem FSS 280
für kommerzielle Anwendungen
2 Laufwerke
mit je 640 kB formatiert
im Gehäuse mit eigener Stromversorgung incl.
erphi
Autopatch-Controller AFDC 3
Handbuch und Diskette mit Dienstprogrammen
Verkaufspreis incl. MwSt.
DM 1.698,–

Noch universeller und noch komfortabler...

Betriebssysteme
DOS 3.3
DiversiDOS 2-c, 4-c,
ProDOS 1.0.1, 1.1.0, 1.1.1
Pascal 1.1, 1.2
CP/M 2.20, 2.23, 2.26

SLOT-Unabhängig

Die bisherigen Vorzüge bleiben erhalten, wie

Autopatch-Boot
automatische Erkennung und Erweiterung der Betriebssysteme während des Bootvorgangs (Betriebssystem auf der Boot-Diskette bleibt unverändert)

Originalsystem-Boot
Von herkömmlichen Apple®-Disketten (unabhängig vom Laufwerksformat) weiterhin möglich

Problemloses Übertragen
herkömmlicher Apple®-Software auf Disketten höherer Kapazität durch SIM 35-Hilfsprogramm (simuliert 35-Spur-Laufwerk unabhängig vom tatsächlichen Laufwerksformat)

erphi electronic
GmbH
Dammweg 3
D-8011 Großhelfendorf
Telefon (0 80 95) 441
Telex 528 021 erphi d

Redakteur

Für unsere Zeitschrift Peeker suchen wir einen Redakteur mit soliden Assembler-Programmierkenntnissen.
Es erwartet Sie eine abwechslungsvolle Tätigkeit in einem großen Verlagshaus.

Ihre Bewerbung mit den üblichen Unterlagen richten Sie bitte an:

Verlagsgruppe Dr. Alfred Hüthig
Personalabteilung
Postfach 10 28 69
6900 Heidelberg

DAS APPLE II HANDBUCH

Die rasche Orientierung für APPLE IIplus, IIe, IIc

Schnelle Antwort auf Alltagsfragen am APPLE II – leicht nachzuschlagen, praxisbezogen, für IIplus, IIe, IIc in nur 1 Buch!
Unterschiede IIplus/IIe, DOS 3.3/ProDOS, E/A-Interfacekarten/Ports, 40/80 Zeichendarstellung, US/DTS-Tastatur, 48K/128K Systeme etc.
Grafik/Soundmöglichkeiten, eine der APPLE-Stärken, in stark erweiterter Beschreibung.
Kurzführer „Steckkartenerweiterungen" mit Fotos; „Sofortbetrieb von Disketten/Cassettengeräten"; „Druckerbetrieb"; „Direktbefehle"; „Tastaturbedienung" etc.

Backgroundwissen: BASIC für Beginner/Professionelle; MC/BASIC-Kombination; MC-Entwicklung mit MONITOR/MINIASSEMBLER; APPLE-PASCAL-BS; Disketten/Plattenspeicherung; Dateiformate etc.
Ausführlicher Anhang zu Editor, Speicherbelegung, Codes des APPLESOFT-Interpreters etc.

DAS APPLE II HANDBUCH für IIplus, IIe, IIc, 472 Seiten, Softcover, DM 66,–

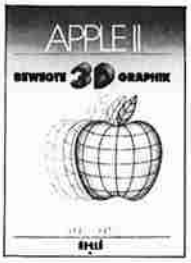
te-wi Verlag GmbH
Theo-Prosel-Weg 1
8000 München 40

Weiterführende Literatur...

NEU

Reparaturanleitung Computer: Apple II, IIplus
Einzigartige Serviceunterläge für Reparaturen und Entwicklungsarbeiten am Apple II. Enthält Schaltpläne, Bauteile- und Vergleichstypenliste; Prüfpunkte mit Oszillogrammen der Signalformen, Logiktabellen, Spannungsangaben; schnelle Servicetests; Anleitung zur systematischen Fehlersuche. In A4-Mappe, DM 29,80

LOGO – Jeder kann programmieren
(Daniel Watt)
Buch des Jahres in den USA. Für die Computer APPLE II, C-64, IBM PC, ATARI bis 520 ST., TI-99 und Schneider CPCs.
Hochwertiges Textbuch für Logo-Kurse für zu Hause und im Lehrbereich.
384 Seiten, A4, DM 59,–



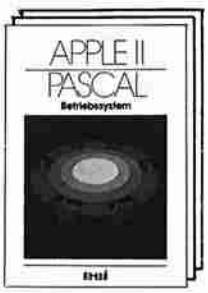
NEU

APPLE II – Bewegte 3D-Graphik
(Phil Cohen)
Selbstentworfene Graphiken und Diagramme – animiert oder als Standbilder – eben oder räumlich: alle erforderlichen BASIC-Programme mit Erklärung finden Sie in diesem Buch.
200 Seiten, Softcover, DM 49,–

Apple Maschinensprache
Für BASIC-Programmierer der einfachste Zugang zur Muttersprache des Apple. Wesentlich schnellere Maschinenprogramme, direkte Manipulation des Mikroprozessors 6502 im Apple – als Brücke dorthin benötigt dieses Buch nur die drei BASIC-Befehle, POKE, CALL, PEEK. D. Inman/K. Inman. DM 49,–

NEU

APPLE WORKS + APPLE II, IIe, IIc = Elektronischer Schreibtischmanager
Dieses Programmsystem vereinigt die Funktion Texterstellung, Datenarchivierung, Formblattkalkulation, Datenfernübertragung. Das System mit den höchsten Verkaufsziffern. Sämtliche System/Anwendungsfragen in 2 Bänden. Beispiele aus der Wirtschaft u. v. m.
je 264 Seiten, je DM 49,–



Erstes deutsches Referenzwerk sämtlicher Befehle und Systemroutinen von Apple II, IIplus, IIe
APPLE II PASCAL NEU
Betriebssystem, 272 S., DM 49,–
Sprache, 216 S., DM 39,–
Pascal 1.2 Addendum, 112 S., DM 36,–

Grundlagenbuch, Bestseller
APPLE II PASCAL,
Eine praktische Anleitung,
544 S., DM 59,–

tm 4439

Noch im Programm:
6502 – Programmieren in Assembler DM 59,–
VisiCalc, 50 Programme auf Diskette, DM 79,–
Computer für Kinder, APPLE II, DM 29,80

In Vorbereitung:
Macintosh Programmier-Handbuch mit Microsoft BASIC 2.0 DM 59,–

Warum wollen Sie ein teures Textverarbeitungsprogramm kaufen, wenn es ein billiges und besseres gibt?

Fast-Writer

von Harald Grumser
Programmdiskette und Handbuch
Gerätevoraussetzung: Apple IIe oder IIc (nicht II+)

DOS-3.3-Version. Auslieferung ab 1.6.86
Normalpreis DM 128,– (ISBN 3-7785-1419-9)
Sonderpreis für Peeker-Abonnenten DM 98,–

ProDOS-Version. Auslieferung ab 1.9.86
Normalpreis DM 128,– (ISBN 3-7785-1421-0)
Sonderpreis für Peeker-Abonnenten DM 98,–
Kombinationspreis für Bezieher der DOS-3.3-Version DM 28,–

Der Fast-Writer von Harald Grumser ist in zahlreichen Funktionen wie Scrollen, Suchen und Ersetzen mit Abstand das schnellste und damit angenehmste Textverarbeitungsprogramm für den Apple IIe oder IIc.

Flexibilität

Viele Textverarbeitungsprogramme sind geschützt und laufen deshalb nur in Verbindung mit normalen Disk-II-Laufwerken. Nicht so der Fast-Writer!

– Der Fast-Writer modifiziert weder DOS 3.3 (oder Diversi-DOS) noch ProDOS und kann deshalb mit BRUN FAST.WRITER gestartet werden. Unter Diversi-DOS ist der Fast-Writer dann in 3 Sekunden im Speicher. Vergleichen Sie einmal, wie lange es dauert, bis andere Textprogramme im Speicher sind!

– Da der DOS-3.3-Fast-Writer in den oberen 16K (= Language Card) liegt, kann man ihn vorübergehend verlassen und mit einem einfachen Befehl wieder starten. Mit anderen Worten: Der Fast-Writer ist permanent verfügbar, auch wenn Sie zwischendurch beispielsweise mit FID Dateien kopiert haben.

– Der Fast-Writer läuft mit allen externen Datenspeichern, die für DOS 3.3 oder ProDOS gedacht sind, z.B. mit dem Erphi-160-Spur-Subsystem, mit der Megaboard-MDB-Festplatte, mit RAM-Karten usw. Spezielle Anpassungen sind nicht erforderlich. Suchen Sie einmal ein Textprogramm, das mit diesen Datenspeichern auf Anhieb funktioniert!

– Der Fast-Writer kann mühelos über ein Menü für Ihre speziellen Aufgaben konfiguriert werden. Sie können z.B. per Knopfdruck die Zeilenbreite (normal 80 Zeichen) am Bildschirm einstellen, wobei ab einer Breite von weniger als 41 Zeichen automatisch auf die größere Bildschirmschrift umgestellt wird. Ferner können Sie die Größe des Arbeitsspeichers (insgesamt ca. 35 500 Zeichen) beliebig in Textspeicher und Hilfspuffer (für Löschen und Blockverschieben) aufteilen. Wenn Sie z.B. große Textblöcke im Speicher zu verschieben haben, so können Sie einen entsprechend großen Hilfspuffer von z.B. 10 000 Zeichen einrichten. Damit entfällt das zeitraubende Zwischenspeichern und Einlesen von Diskette.

Befehlsvorrat

Der Fast-Writer verfügt über eine große Zahl von Befehlen, von denen Sie in der Praxis jedoch nur wenige benötigen. Fünf Befehlsübersichten sind durch eingebaute Hilfsübersichten immer abrufbar, so daß Sie schon nach einer mehrstündigen Einarbeitung auf das Handbuch verzichten können. Eine Auswahl der wichtigsten Befehle:

– Freie Cursorbewegung in allen vier Richtungen mit eingebauter Schnell-Scroll-Routine.

– Diverse, per Knopfdruck ein- und ausschaltbare Optionen, z.B. Wortumbruch/kein Wortumbruch, Return sichtbar/unsichtbar, Kopfzeile (Statuszeile) mit Speicherbelegung, Cursorposition usw. eingeblendet/ausgeblendet, Bildschirm geteilt/ungeteilt, Tabulatorleiste sichtbar/unsichtbar, überschreibmodus (statt normalen Einfügmodus) ein/aus usw.

– Eingabe von Kontrollbuchstaben (einschließlich Ctrl-V!) möglich. Automatische Konvertierung in Groß- oder Kleinschreibung (unter Berücksichtigung der Umlaute und ß!)

– Extrem schnelles Suchen und Ersetzen von Zeichenketten (vorwärts und rückwärts).

– Makros frei definierbar und per Knopfdruck abrufbar. Makros können nicht nur stereotype Wortfolgen sein (z.B. „Sehr geehrte Herren"), sondern auch alle Befehlsfolgen, die man beim Fast-Writer sonst über die Tastatur eingeben würde. So läßt sich beispielsweise ein Text automatisch von Laufwerk 1 laden und auf Laufwerk 2 speichern.

– DOS-Kommandos wie Catalog, Delete, Rename usw. immer verfügbar (bei DOS 3.3 zusätzlich Init, bei ProDOS zusätzlich Online und Datum)

– Ausdruck auf Matrixdrucker (normal endlos), Schreibmaschine (normal mit Einzelblatt) und zu Kontrollzwekken auf Bildschirm; links- und rechtsbündig, zentriert und Blocksatz; einstellbarer linker, rechter und oberer Rand (im Text änderbar), bei Bedarf mit Kopfzeile und Paginierung usw. Der Ausdruck kann über eigene Druckertreiber umgelenkt werden, um z.B. Probleme mit Typenrädern, Steuerzeichen usw. zu beheben.

– Makros, Druckparameter, Druckertreiber und Tabulatoren können auf Diskette gespeichert werden.

Hüthig Software Service · Postfach 10 28 69 · 6900 Heidelberg 1